

ROV BOTS • SERVICE BOTS • BOTS BUILDING BOTS

SERVO

FOR THE ROBOT INNOVATOR

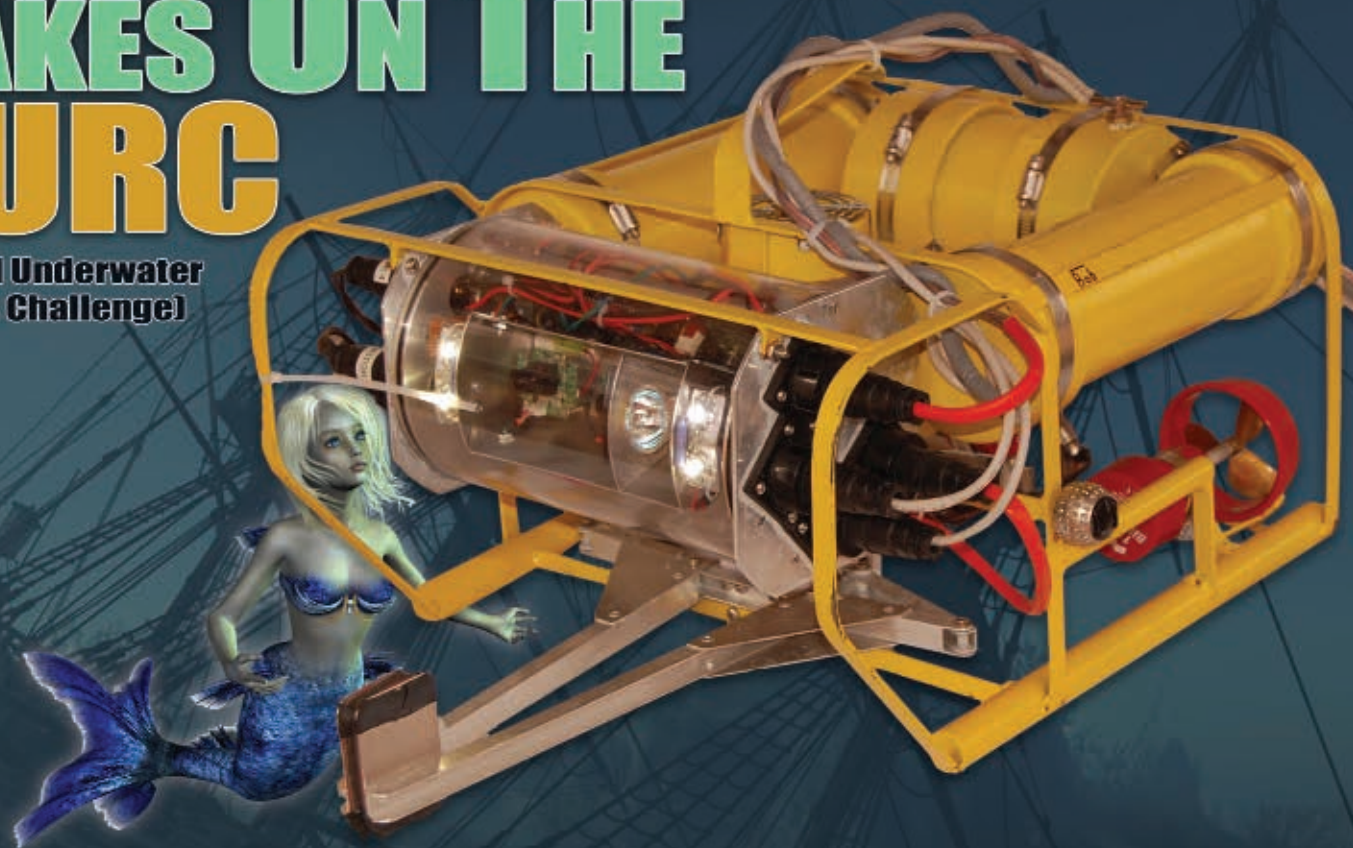
www.servomagazine.com

MAGAZINE

February 2010

notBob ROV Robot TAKES ON THE NURC

(National Underwater
Robotics Challenge)



- ◆ Robot Base On A Budget
- ◆ Building Robot Parts With A CNC Laser
- ◆ CheapBot Smart Proximity Detector
- ◆ Build A Robot From An Old VHS Tape
- ◆ New Bioloid Premium Kit From Robotis

U.S. \$5.50 CANADA \$7.00



The Evolution Continues

BIOLOID

PREMIUM Kit

New Release
Special Gift



Zigbee module
for 1,000 customers

All-in-one package for
advanced robot builders



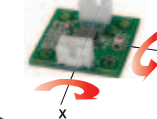
CM-510
Main Controller



RoboPlus



Gyro Sensor
Walk Balancing



DMS
Distance Measurement



Humanoid Skin



RC-100
Remote Controller



Li-Po Battery
11.1V 1000mA



USB2Dynamixel



Dynamixel AX-12+
18 DOF Actuators



IR Sensor
Object Detection



- Excellent humanoid walking performance (Adjusting posture while walking)
- Various sensors including Gyro, Distance, IR and more external ports
- Remote control capability (IR-default, Zigbee-optional)
- C-style programming & motion teaching with RoboPlus S/W (USB interface included)
- Transparent humanoid skin for customization
- Digital Packet communication with daisy chain topology
- Building various robots using versatile expansion mechanism

ROBOTIS

www.robotis.com

TEL : +82-70-8671-2600

Email : contactus2@robotis.com



Robots, lots of robots!

The World's Leading Source
for Domestic and Professional Robot Technology



www.robotshop.com

Columns

08 Robytes

by Jeff Eckert

Stimulating Robot Tidbits

10 GeerHead

by David Geer

Artificial Intelligence Brings
Humanoid Robots to Life —
Part Deux

13 Ask Mr. Roboto

by Dennis Clark

Your Problems Solved Here

79 Then and Now

by Tom Carroll

A Brief Look at Service Robotics

The Combat Zone...

Features

24 MANUFACTURING:

Hobby Guide to Metal
Lathe Work: Part 1

27 PARTS IS PARTS:

Hole Drilling Redux

Events

27 Results/Upcoming Events

28 EVENT REPORT:

Mayhem in Miami

30 ROBOT PROFILE:

3pd



Service Robotics
PAGE 79



Departments

06

Mind/Iron

16

New Products

18

Events Calendar

19

Showcase

20

Bots in Brief

64

SERVO Webstore

82

Robo-Links

82

Advertiser's

Index

In This Issue ...



PAGE 43

58 Building Robot Parts with a CNC Laser

by Jose Quinones

A CNC laser is a powerful and highly versatile tool. Learn some of the advantages, disadvantages, and basic techniques to have *this* machine build *your* machine.

67 Base on a Budget

by Chris Savage

Get a useable robot platform up and running with only a minimal budget.

32 The CheapBot Smart Proximity Detector

by L. Paul Verhage

This proximity detector incorporates a PICAXE-08M so that *it* takes care of detecting obstacles and leaves the driving to the robot controller.

36 The μ OLED-128

by Fred Eady

High-quality color graphics for your mechatronic creation are just a serial port away. Microchip's new P32MX microcontrollers have six serial ports to choose from. This month we'll use one of them to drive a 4D Labs μ OLED-128-G1 Display Module.

43 The Bioloid Premium Kit

by Rob Farrell

Learn the details of ROBOTIS' latest offering in the humanoid league.

48 The Cassette Bot

by Bryan Jackson

Remember VHS cassette tapes? Well, now you can turn them into robots!

52 The Typewriter Repairmen 2009 NURC ROV: notBob

by David Forbes, Steven Forbes, Jim Forbes, and Janet Forbes

Follow this family team's build of a submersible robot that captured the overall winner award at the 2009 NURC competition.

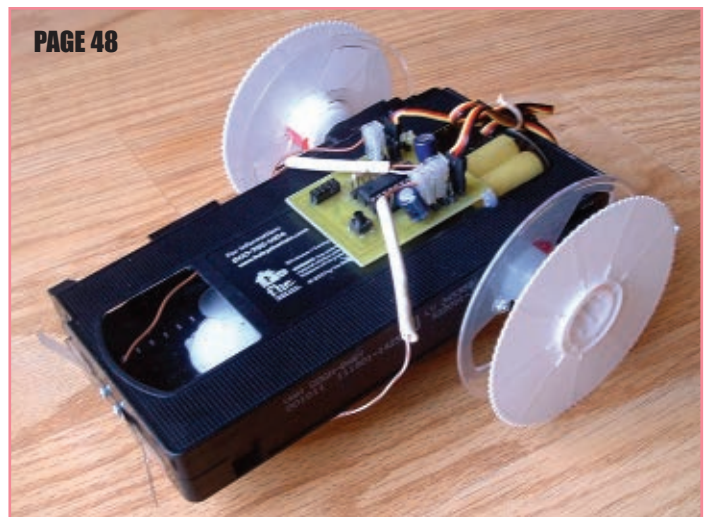
72 Teach Your Old RoboScout Some New Scripts

by Robert Doerr

See how using an optional PC interface can let your robot learn some different tricks.



PAGE 48



Mind / Iron

by Bryan Bergeron, Editor



Building on the Fundamentals

If you pick up a copy of the *IEEE Transactions on Robotics*, the first thing you'll likely notice is the relative abundance of mathematics, relative to the predominance of prose and photos in *SERVO*. This difference reflects the goals of the readership of each publication. An enthusiast doesn't need to know the stress on the gear train within a servo as long as the unit as a whole will handle a particular load. For an engineer designing a servo mechanism, however, these details matter. Moreover, the nature and magnitude of the stress on the gear train is most completely and unambiguously expressed in mathematical terms.

The reason for comparing the two publications is that they represent different ends of the spectrum of robotics. You may be content with kit building and working with pre-architected robots. On the other hand, you may have your

sights set on a career designing new robotics platforms, sensors, and intelligent computing systems. If the latter case describes you, then you'll need to study the fundamentals of robotics, from statics and dynamics to strength of materials and biomechanics – either in a formal college program or on your own.

If you opt for a traditional university education, then you'll probably have to enroll in an electrical engineering or mechanical engineering program and fill your electives with robotics-related courses. There are very few true undergraduate robotics engineering programs in the world. For example, Worcester Polytechnic Institute (www.wpi.edu) established the first bachelor's degree program in robotics in 2006.

Going it alone is possible, but only if you have discipline and time to study. And there's really no such thing as going it alone. You'll have to find a mentor with similar interests to get the most out of your studies. There's nothing as frustrating as studying with the nagging feeling that if you only had a real teacher, you'd progress more rapidly.

Regardless of your path, you can and should start working on the fundamentals now. The most fundamental building block in robotics is mathematics – algebra, trigonometry, statistics, differential equations, and calculus. There's an abundance of texts on Amazon and elsewhere. If you're more at home with online computer-aided instruction and serious games for learning math concepts, then there are a number of free sites on the web, including Wolfram MathWorld (mathworld.wolfram.com) and Math.com (www.math.com).

If your eyes automatically glaze over at the sight of a math textbook, consider picking up a copy of *The Manga Guide to Calculus*, published by No Starch Press (www.nostarch.com), 2009. Other than reading from front to back and depicting unadventurous costumes, the book resembles every other Japanese manga book marketed in the US. The paperback isn't exhaustive, but it provides a painless introduction to calculus within the context of a story told in comics. Once you've mastered the book, you can move on to more traditional treatments.

By the way, No Starch Press also offers a manga treatment of electricity, *The Manga Guide to Electricity*, which is a fun introduction to electricity for high school level students. It's appropriate for a budding roboticist who may be mechanically talented but has yet to grapple with basic concepts such as Ohm's Law.

Good luck on your path. **SV**

STEER WINNING ROBOTS WITHOUT SERVOS!



Perform proportional speed, direction, and steering with only two Radio/Control channels for vehicles using two separate brush-type electric motors mounted right and left with our **mixing RDRF dual speed control**. Used in many successful competitive robots. Single joystick operation: up goes straight ahead, down is reverse. Pure right or left twirls vehicle as motors turn opposite directions. In between stick positions completely proportional. Plugs in like a servo to your Futaba, JR, Hitec, or similar radio. Compatible with gyro steering stabilization. Various volt and amp sizes available. The RDRF47E 55V 75A per motor unit pictured above. www.vantec.com

VANTEC

Order at
(888) 929-5055

Published Monthly By
T & L Publications, Inc.
 430 Princland Ct., Corona, CA 92879-1300
(951) 371-8497
 FAX **(951) 371-3052**
 Webstore Only **1-800-783-4624**
www.servomagazine.com

Subscriptions
 Toll Free **1-877-525-2539**
 Outside US **1-818-487-4545**
 P.O. Box 15277, N. Hollywood, CA 91615

PUBLISHER
 Larry Lemieux
publisher@servomagazine.com

**ASSOCIATE PUBLISHER/
 VP OF SALES/MARKETING**
 Robin Lemieux
display@servomagazine.com

EDITOR
 Bryan Bergeron
techedit-servo@yahoo.com

CONTRIBUTING EDITORS
 Jeff Eckert
 Tom Carroll
 Dennis Clark
 Fred Eady
 Rob Farrell
 Robert Doerr
 Bryan Jackson
 David Forbes
 Steven Forbes
 Chris Williamson
 Joey Daoud
 Jenn Eckert
 David Geer
 R. Steven Rainwater
 Kevin Berry
 Paul Verhage
 Jose Quinones
 Chris Savage
 Jim Forbes
 Janet Forbes
 Nick Martin

CIRCULATION DIRECTOR
 Tracy Kerley
subscribe@servomagazine.com

**MARKETING COORDINATOR
 WEBSTORE**
 Brian Kirkpatrick
sales@servomagazine.com

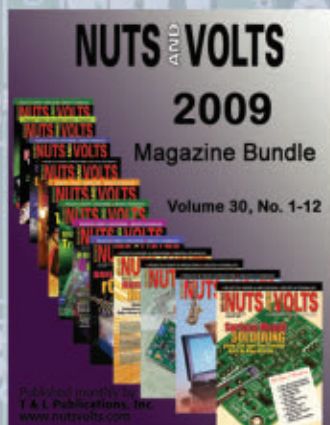
WEB CONTENT
 Michael Kaudze
website@servomagazine.com

ADMINISTRATIVE ASSISTANT
 Debbie Stauffacher

PRODUCTION/GRAPHICS
 Shannon Christensen

Copyright 2010 by
T & L Publications, Inc.
 All Rights Reserved

All advertising is subject to publisher's approval. We are not responsible for mistakes, misprints, or typographical errors. *SERVO* Magazine assumes no responsibility for the availability or condition of advertised items or for the honesty of the advertiser. The publisher makes no claims for the legality of any item advertised in *SERVO*. This is the sole responsibility of the advertiser. Advertisers and their agencies agree to indemnify and protect the publisher from any and all claims, action, or expense arising from advertising placed in *SERVO*. Please send all editorial correspondence, UPS, overnight mail, and artwork to: **430 Princland Court, Corona, CA 92879.**



NUTS & VOLTS
2009
 Magazine Bundle
 Volume 30, No. 1-12

Choose your preferred format:
 print or digital media.

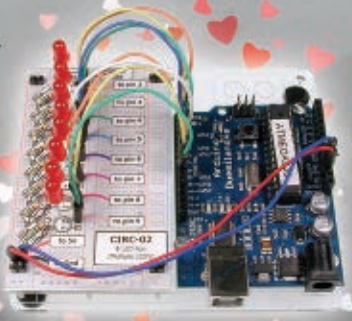
We have just what you're looking for!

12 magazine bundle **\$57.00**
 12 issues on CD-ROM **\$24.95**
 Order today 800 783-4624
 or online @ www.nutsvolts.com

Italian: The language of Love ...and microcontrollers

So this Valentine's Day, whisper in the ear of your loved one: "Ti amo, mio piccolo Ardweeny". She'll understand you want the Free Ardweeny that comes with each ARDX Oomlout Arduino Experimenters's kit.





This image is to scale (...no, seriously!)

<http://www.solarbotics.com/products/ardy/> SKU: ARDY Price: \$84.95



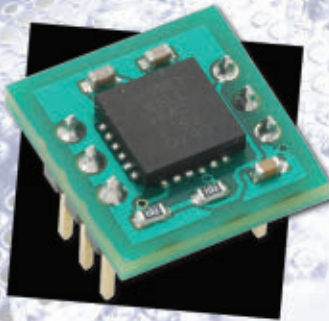
SOLARBOTICS

www.solarbotics.com 1-866-276-2687

PS- Or is French the Language of Love? Whichever it is, we're pretty sure that the other one is the Language of Lunch. It's so hard to keep track of these things.



HMC6352 Compass Module

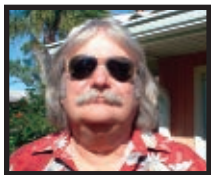


The **HMC6352 Compass Module** (#29323; \$29.99) is fully integrated and combines 2-axis magneto-resistive sensors with the required analog, digital, microprocessor and algorithms required for heading computation. It provides a direct heading angle and includes internal calibration algorithms within its firmware. Stray magnetic field protection and temperature compensation are also included.

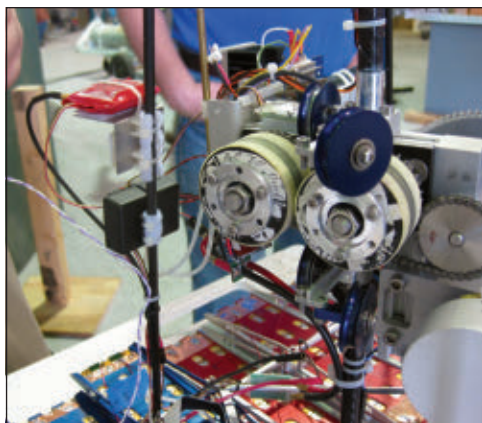


www.parallax.com

Order the **HMC6352 Compass Module** (#29323; \$29.99) at parallax.com or call 888-512-1024 (Mon-Fri, 7 a.m. - 5 p.m., PST).
 Propeller, Parallax, and the Parallax logo are trademarks of Parallax Inc.
 Prices subject to change without notice.



Climber Wins \$900k



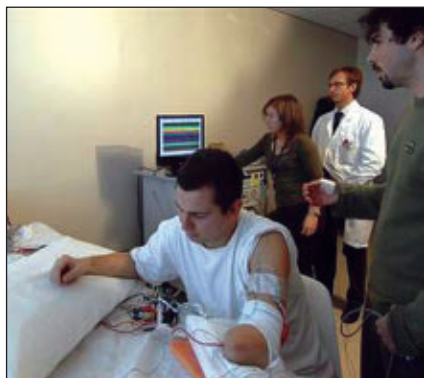
Close-up of the climber's cable-gripping wheels, reportedly improvised from an in-line skate. Courtesy of www.spaceelevatorblog.com

It's been nearly 40 years since Jack Bruce composed "Rope Ladder to the Moon," but what once was a psychedelic musing has turned out to be somewhat prophetic. Something similar was recently demonstrated at the Dryden Flight Research Center by LaserMotive (www.lasermotive.com) which won \$900,000 from NASA's Centennial Challenges program. The company's

laser powered robotic climber crawled up a 900 m (2,953 ft), quarter inch cable, suspended from a helicopter, in three minutes, 48 seconds — easily beating the Level 1 standard of 7.5 minutes. The demonstration is a step toward validating the concept of a space elevator that can lift humans and cargo into the sky without the need of rockets and such. (In a real space elevator, the cable would be suspended by inertial force rather than a helicopter.)

The LaserMotive system uses a high power laser array to shine ultra-intense infrared light onto solar cells which convert the light into electric power to drive its motor. It's noteworthy that the bot did not meet the Level 2 standard of three minutes or less, so it did not earn the entire \$2 million prize. The remaining \$1.1 million can still be claimed, so get ye to the drawing board.

Robotic Hand Passes Test



Last December, a group of European scientists based at the University Campus Bio-Medico of Rome (www.unicampus.it) announced that they had successfully connected a

In the LifeHand experiment, and amputee manipulates a biomechanical hand purely via mind control. Courtesy of Bio-Medical Campus University.

robotic hand to a man who had lost an arm in a car accident, allowing him to control the prosthetic with his thoughts and to actually feel sensations in the artificial limb. Reportedly, the \$3 million LifeHand experiment marked the first time an amputee has been able to make complex biomechanical movements using his mind to control a hand connected to the nervous system.

Reportedly, during the month that 26 year old Pierpaolo Petruzzello was connected, he learned to wiggle the robotic fingers independently, make a fist, grab objects, and make other movements. One of the research team's neurologists revealed, "Some of the gestures cannot be disclosed, because they were quite vulgar." We therefore can assume that Mr. Petruzzello either has a sense of humor or a lack of gratitude. Details have been published in the Proceedings of the National Academy of Sciences (www.nasonline.org).

Narcissists Rejoice!



Robotist Hiroshi Ishiguro putting the moves on himself.

If you're in the habit of staring endlessly at your reflection in pools of water, or are otherwise just in love with yourself, catch the next flight to Japan and drop into one of the Sogo & Seibu department stores, where you can order a robotic duplicate of yourself. Built by Kokoro (www.kokoro-dreams.co.jp) — known for its Actroid fembots — the silicone face and body shape will match your own down to details like hair and eyes. It can duplicate your movement characteristics, and it will even speak in your own voice (synthesized from recordings). Reportedly, though, this model cannot walk or perform any

other below-the-waist functions you might be imagining, so the entertainment possibilities for your spouse are limited. However, it can sit there and say "yes, dear" and even pretend to enjoy watching Oprah, which alone might make it worth the \$225,000.

Rug Sucker Uses Laser Vision

By all accounts, iRobot (www.irobot.com) has sold in excess of 3,000,000 of its Roomba robotic vacuums since 2002, with nothing much appearing in the way of serious competitors. But Neato Robotics (www.neatorobotics.com) — a Silicon Valley startup — is upping the ante with its XV-11 all-floor



The XV-11 robotic vacuum. Courtesy of Neato Robotics.

vacuum system: it is the first to employ laser vision to map floor space and avoid obstacles that other cleaning bots detect only by impact. Using Neato's Room Positioning System (RPS) Technology™, the XV-11 takes a 360° peek at a room, taking note of walls, furniture, doorways, and any obstacles located at up to 4 m away. It then uses simultaneous location and mapping (SLAM), and proprietary path-planning algorithms to clean the space in a back-and-forth pattern rather than the seemingly disorderly movements of the Roomba. The result is said to be faster and more power-efficient cleaning. Other features include higher suction, a low profile (<4 inches in height), a larger dirt bin (660 cc, or 0.7 ft³), and self-docking for recharge. You can pick one up for \$399, including a 30-day, money-back guarantee and a one-year hardware warranty.

Robot Conference Coming Up

If you've always wanted an excuse to visit Alaska on the company expense account, consider the 2010 IEEE International Conference on Robotics and Automation (ICRA2010) to be held in Anchorage, May 3 through 8. The conference theme is "50 Years of Robotics," reflecting on achievements of the field and the broad impact of robotics and automation research, development, and education. It's too late to submit a paper of your own, but you can sit in on invited sessions, view videos, and attend tutorials and workshops, plus there will be an exhibit site for companies, universities, publishers, and researchers. Some 1,300+ attendees are expected. Rooms are available at the Hilton for a group rate of \$145/night and the Marriott at \$155/night. (Of course, better deals are available online.) You don't have to worry about freezing: May temperatures in Anchorage average 39 and 55°F; low and high. Info is available at icra2010.grasp.upenn.edu. **SV**

2009 CD ROM



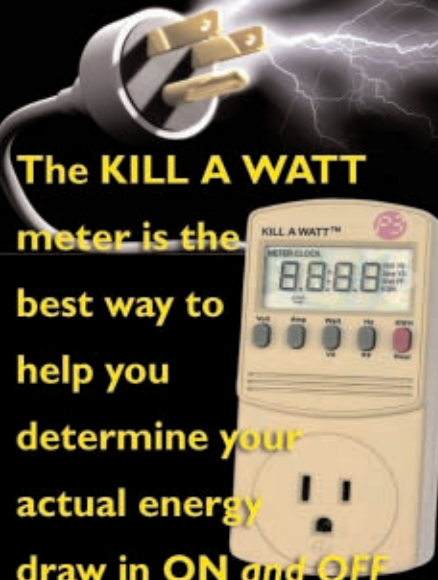
On Sale Now!

Contents include

- January through December 2009 issues of **SERVO Magazine**
- Supporting code and media files from each issue
- Windows compatible Adobe Reader
- MAC compatible Adobe Reader

Only \$24.95

"THE PHANTOM DRAW"



The KILL A WATT meter is the best way to help you determine your actual energy draw in ON and OFF home appliances.

To order call 1 800 783-4624 or online www.nutsvolts.com

\$29.95 plus S&H



Artificial Intelligence Brings Humanoid Robots to Life (Part Deux)

It is not so much the practicality of humanoid robots that drives interest in their development as it is the familiarity, sources say. "People are interested in things that are anthropomorphic. This makes interaction with [humanoid] robots easier," says James J. Kuffner, PhD, associate professor of the Robotics Institute at the School of Computer Science, Carnegie Mellon University. So, while there are many practical robots taking shape as imitations of other biological life forms (such as snakes for pipeline examination), a place has been indelibly etched for humanoid robots.

In developing humanoid robots, most researchers imitate people's useful functions and behaviors in their work in order to replicate skills that will be useful for our society, according to Kuffner.

"Some researchers are interested in [going beyond that and] making a clone of someone that is indistinguishable from a human being. That is a much harder thing to do. And for robots that we simply want to improve our society, it is not

necessary to have a machine that looks exactly like a human being," Kuffner explains.

In fact, it may even be counter-productive. One example is something called the "uncanny valley." The uncanny valley is an interesting phenomenon that was discovered several decades ago in which people have an affinity for things that are anthropomorphic, Kuffner continues.

It has to do with our neurons. By making a machine anthropomorphic, we can make the interaction with the robot easier and give people a natural appeal towards the machine.

"But something funny happens when the machine gets to be so realistic that it starts to become indistinguishable from real humans. Then people are not sure whether it really is human or not. And that's called the uncanny valley in terms of if you plot an affinity for the machine with a vertical axis and its degree of realism with a horizontal axis, people start to like it the more anthropomorphic it is and then all of a sudden it reaches a point where people are afraid, they are not sure what it is and they are scared of it [and the affinity for the machine on the vertical axis drops] and they have a strong aversion to them," says Kuffner. A lot of psychologists think it is the fear of zombies or dead bodies, according to Kuffner.

If people can't tell the difference between a

Humanoid robot in the CMU Robotics Institute lab exhibiting a grasping technique.



robot and a real person, they become hesitant to interact with it. "I try to create robots with a humanoid shape but that are clearly not human, because we are worried about that uncanny valley," says Kuffner.

When looking at the things we can get a humanoid to do, it would be quite difficult, for example, to leap from a singularly tasked industrial robot to a humanoid that could perform all the functions necessary to build an automobile from the ground up given the tools and parts, Kuffner notes. A sufficiently skilled mechanic (or two) might be able to do this.

However, because robots are computers and computers are great at searching against a database, it is highly conceivable that humanoids can quickly become skilled at tasks where optimal choices can be discerned from a list in a database.

Dr Kuffner is optimistic about the advancement of robots using human tools. "The technology is improving a lot in the last decade or so in terms of perception and sensing. In terms of high resolution cameras, laser scanners are now coming on line. Outdoor robots didn't work at all until we had reliable GPS," comments Kuffner.

As these perception capabilities and computing powers improve, using human tools — having a humanoid pick up a cleaning instrument such as a towel and wipe down a surface — could happen in 10 years, Kuffner believes. There are already prototypes that do that.

Kuffner's Work on ASIMO

Kuffner has been working with ASIMO to achieve efficient walking. To accomplish this, he has been developing a footstep planning algorithm that searches possible footstep choices over and over for the best choice to avoid slipping, tripping, and bumping into objects while walking a route to a desired location.

The algorithm looks at the state of the floor and searches all the possible actions [steps], then selects a sequence of actions to take to reach the goal, Kuffner explains. "Imagine you are on rough terrain and you are humanoid and you want to climb to the top of a hill," Kuffner continues.

The humanoid would need to plan a sequence of footstep locations to walk on to reach that goal. The footstep planning algorithm looks at the reachable region of ground for each

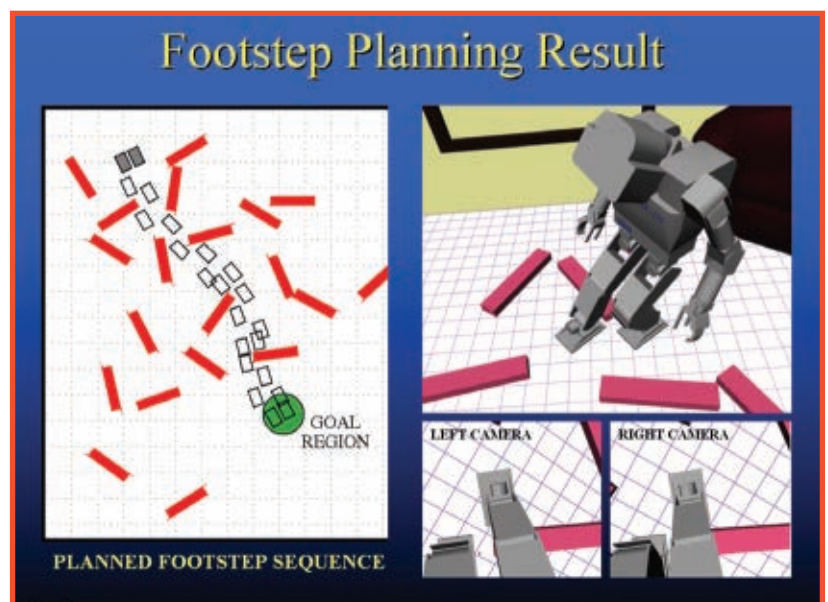


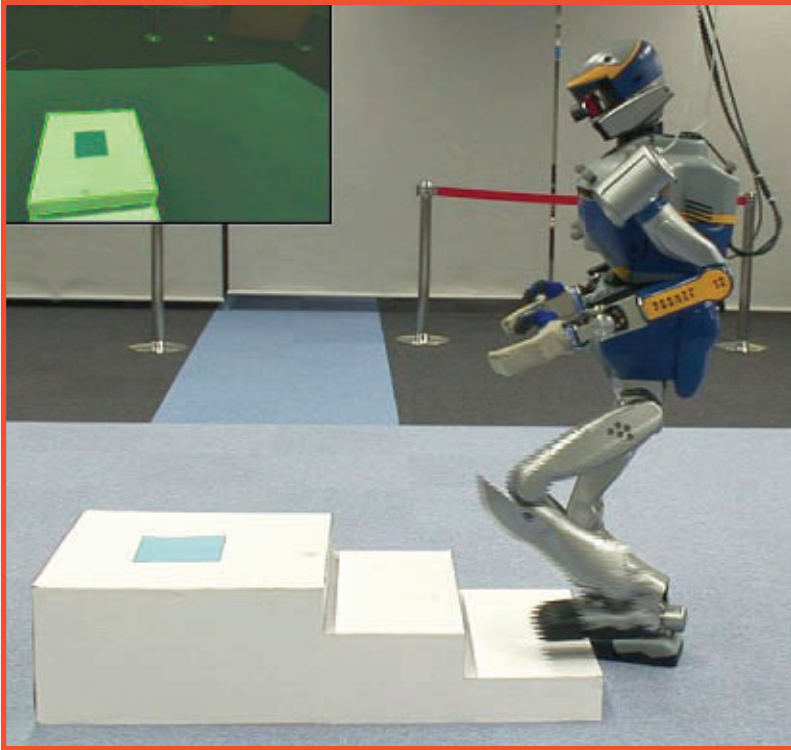
This 3D image shows ASIMO scanning the floor in front of him, searching and discovering the best footsteps to achieve the desired goal.

foot. It processes tens of thousands of possible footstep locations per second. The footstep location that is the most highly safe and highly effective toward reaching the ultimate goal would be selected.

The algorithm also uses heuristics and machine learning so that every time the robot learns a bad place to step, it won't step into a similar place in the future, according to Kuffner. The same algorithm has since been applied to

A 3D drawing of a humanoid using a footstep planning algorithm and images from a right and left camera to plan a sequence of steps toward the goal location.





Humanoid taking a step, up a step.

other robots.

Advances in computing hardware will advance the intelligence of tasks powered by search. "If we accept the possibility that by using search we can achieve human level performance on some tasks, one of the great ways to boost search performance is with massive parallelism," says Kuffner.

With the advent and advances in multicore

processors, more processing can be done in parallel. This will allow computers to search through many possible choices for the robot — whether steps or gestures or other movements — in less time.

Up Close and Personal

Humanoid robots are being groomed for intimate service in society. Topping the list of the most promising applications are assistive health care applications such as lifting and carrying the bed-ridden or helping patients to exercise properly and regain lost motor control, comments Stefan Schaal, associate professor, Computer Science and Neuroscience, USC.

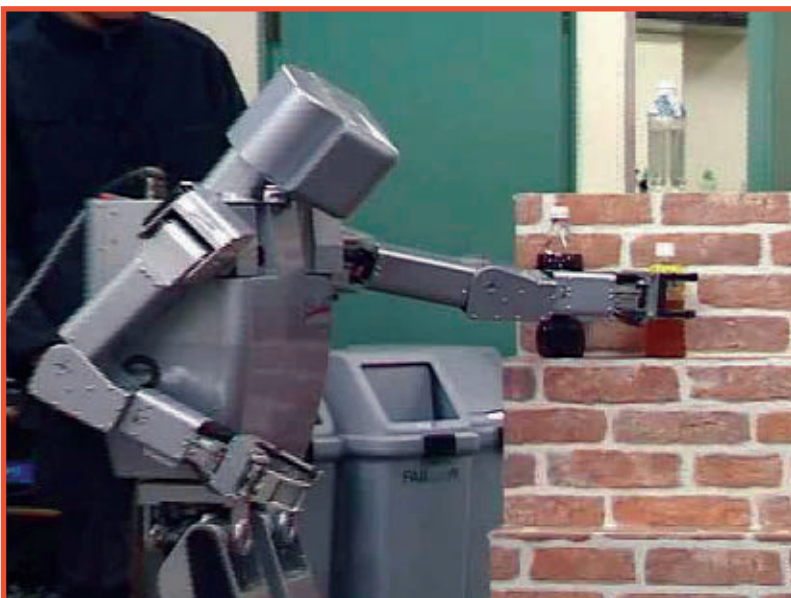
Schaal believes these types of robots will be commercially available in 10 to 20 years. The idea is that you would simply turn the robot on and it would mimic human behavior related to physical therapy.

There is a significant amount of research going into using humanoid robots to help educate autistic children. "Having a really [humanly] physically embodied robot is very appealing to these children," says Schaal.

To make humanoids behave acceptably in society with appealing behaviors and gestures, researchers in media labs are using behaviors that make people appealing. There are stereotypic behaviors that people exude that all people take in certain ways, according to Schaal. "It is very easy to exhibit a bad gesture that repels people," Schaal says.

Conclusion

Intelligence that computers possess today can be leveraged to form the AI that robots will have tomorrow. By applying the skills that computers are good at now — such as searches to select the best behavior choices for a given task — researchers will develop an increasing storehouse of repeatable, intelligent human behaviors that will be useful in society. **SV**



Humanoid reaching and grasping using perception.

Resources

1) Video about Kuffner's work on ASIMO on YouTube:
www.youtube.com/watch?v=0dv0wJDwtuw

2) Videos linked from the CMU Robotics Institute research lab page:
www-old.ri.cmu.edu/labs/lab_76.html



Our resident expert on all things robotic is merely an email away.
roboto@servomagazine.com

Tap into the sum of *all human knowledge* and get your questions answered here! From software algorithms to material selection, Mr. Roboto strives to meet you where you are — and what more would you expect from a complex service droid?

ASK MR. ROBOTO

by
Dennis Clark

In December, I got some very interesting questions that will take longer to answer than I had time to deal with just before the holiday season. Fear not! Mr. Roboto is on the hunt and will return with answers to those particular questions in future columns! However, some were questions that I could answer with information I had on hand, so we'll discuss those.

This question came in response to a project on my own website at www.techtoystoday.com/projects/botproj.htm.

I'm attempting to design a simple and inexpensive set of tactile feelers for our high school robotics class. I liked your simple approach, but have a question.

We're using the simpler PICAXE processor but the implementation would be comparable, to running a wire to the PICAXE input pin, then pulling the pin high when the switch is closed.

My question is have you had any problems with shorting out the processor? My concern is with the "whiskers" since contacts are bare wires exposed to the environment. I'm concerned that the students will inadvertently fry the processor with static electricity or even something else. Thanks again.

— Scott

You are correct. Without a little more additional circuitry, your microcontroller will be at risk from static electricity. **Figures 1 and 2** show the front and back of the whisker switch to which this question refers. It is dead simple to construct and works really well, with some additions.

It takes a pretty strong soldering iron to solder the piano wire (use small gauge "springy" wire) to

the copper plating of the generic, single-sided un-etched PCB, but it works well if you are patient.

To protect your microcontroller, you should put some kind of ESD (electrostatic discharge) protection in series between the whisker switch and the I/O pin of your processor. I typically use 100 ohms as my guard resistor. A small value will work to take a lot of the "zing" off a static zap and won't slow the signal pulse down much. However in this case, you do not have a fast changing pulse from a flexing wire so even a 1K resistor should work fine. Also, I recommend that you pull the I/O line high and when the whisker flexes, it connects the I/O line to ground. Put the pull-up resistor on the whisker side of the series "anti-static" resistor so that you don't end up creating a voltage divider between the series resistor and the pull-up resistor. Use any value of pull-up between 4.7K and 10K. **Figure 3** is a simple schematic explaining all this.

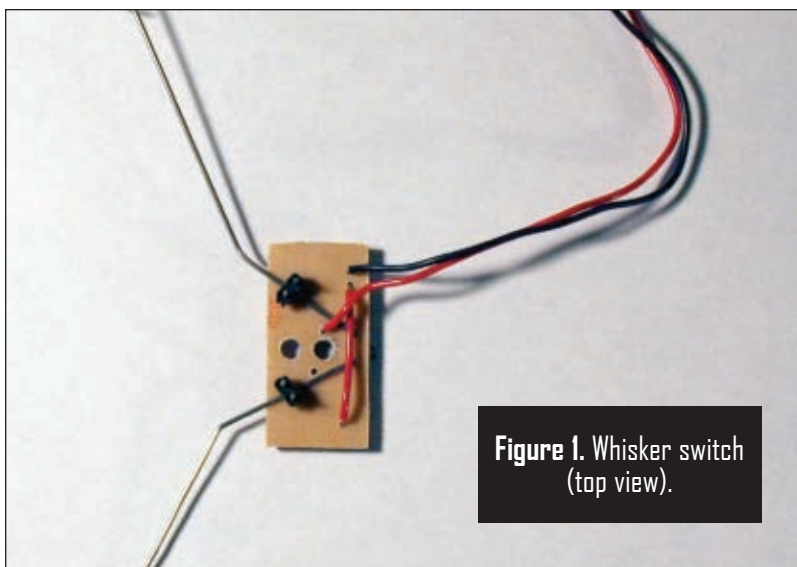


Figure 1. Whisker switch (top view).

Q. I want to start using vision with my robotics projects. There seems to be two camps of robot vision: a CMUcam with their style of simple “blob” recognition and tracking; or a full blown PC with fancy software. Both of these systems for vision are expensive. Is there a way I can get advanced vision that is affordable to the hobbyist and easy enough to understand that I don’t need to be a grad student to use it?

— James

A. Well, if we leave out the stand-alone cameras that allow a limited form of robot vision like the CMUcam and their ilk, we’ll need to use higher level language vision support along with using a PC (to handle the heavy lifting). One way to do vision on your computer and robot is to use Intel’s Open CV. Open CV is an open source computer vision library designed by Intel to run on Intel platforms to give the ability to implement a variety of computer vision capabilities. I have not used this library yet, so I can’t offer any insight on how easy it is to implement. However, it is open source which means it is free — which satisfies one of your chosen requirements. You can find more information about Intel’s OpenCV at <http://software.intel.com/en-us/articles/intel-integrated-performance-primitives-intel-ipp-open-source-computer-vision-library-opencv-faq/>.

This is Intel’s own FAQ site for OpenCV. Of course, there is an open source project on SourceForge for OpenCV at <http://sourceforge.net/projects/opencvlibrary/>.

Like I said, I can’t vouch for how easy these are to get up and running, but if you are interested in “free,” this is a good place to start. One person in our robotics club got a face recognition program to work using this library and it was very cool. Incidentally, this is an Intel library not a Windows library, so Linux and Macintosh are not left out! You just need to be running on Intel processor hardware.

Recently, a second option became available and I got to see a demonstration of this software at our robot club. The package is called RoboRealm and you can find out more about it at www.roborealm.com.

RoboRealm is a very nice bundle that allows you to experiment with a variety of vision algorithms. There are a LOT of these algorithms! RoboRealm allows you to experiment with color tracking, object recognition, edge detection, range finding, and the list goes on. This assortment is very comprehensive. So, what do you do when you see something? RoboRealm can be scripted and has the ability to pass information to serial ports, network ports, http, disk, ftp, and email. Its capabilities open up a lot of areas for exploration.

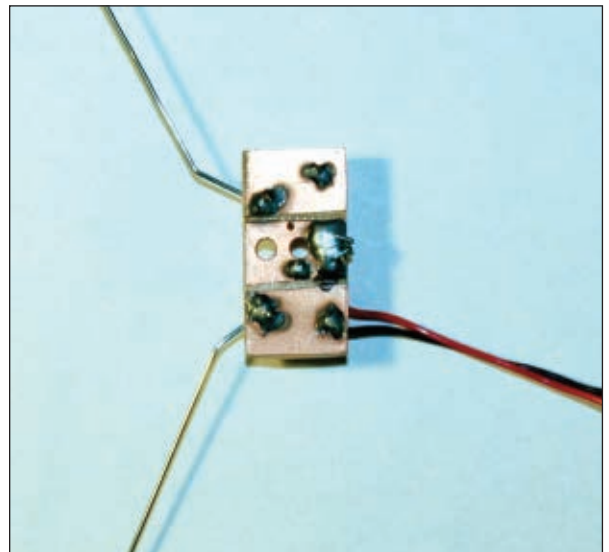


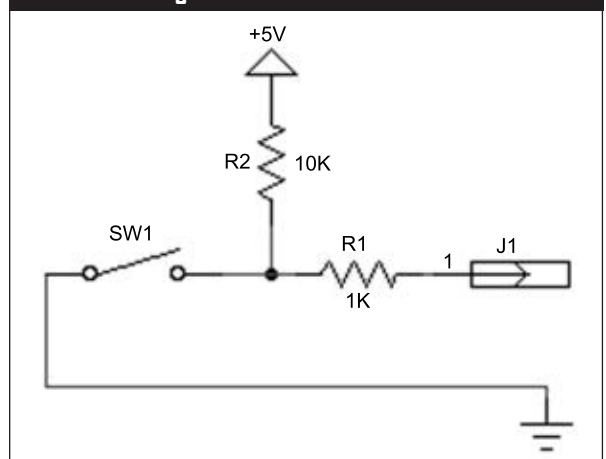
Figure 2. Whisker switch (bottom view).

Demos that I saw had robots aiming at a ball that was moving and when it stopped, the robot shot a rubber band at the ball. Another demo had the robot chasing a ball by using its color and sending motor control values to the robot controller via RS-232 port. Another robot could recognize shapes printed on cardboard flash cards, allowing it to be programmed via the cards while it was running.

What is the downside? RoboRealm is very configurable which means that it can get confusing dealing with all of the options available. Fortunately, the company has lots of tutorials on their website. They are very friendly and answer questions with enthusiasm. RoboRealm is not optimized for any particular task and your frame rate will depend on how fast your computer is, as well as your chosen screen resolution. The smaller your screen resolution, the faster the frame rate.

It has been suggested that you can experiment easily with RoboRealm and various vision tasks,

Figure 3. Whisker schematic.



and when you find something that you like you could program it up using OpenCV optimized for your robot. Experimenting with options is something that RoboRealm does really well.

There is one more down side (to us non-Windows users), however, RoboRealm is currently a Windows software package only. You can run it on a virtual machine such as Parallels or VMware, where it works just fine.

The best part of this software in my opinion, is you can get RoboRealm free for a 30 day demo. Or, you can buy it for \$89. This is a great price for robot hobbyists.

Figure 4 shows yours truly experimenting with tracking a green ball. The little blue square shows what RoboRealm thinks is the object that it is tracking. The data written next to it is the coordinates of the ball. This data can be sent to a robot in a variety of ways for the robot to track it. Very cool.

You can also see in **Figure 4** that there are lots of modules to choose from. Check out the website, download the demo, and have fun. In my opinion, this is a must-have program for your toolbox.

I plan on doing a much more in-depth article using RoboRealm in the future to track and hunt other robots on the Critter Crunch battlefield. This



Figure 4. RoboRealm in action.

was just the program I was looking for to handle a simple vision system to track other robots.

Keep those questions coming and I will do my best to answer them! As usual, you can reach me at roboto@servomagazine.com. Keep on building robots! **SV**

CrustCrawler

Robotics

"Build Smarter."



AX-12 Feedback

- ✓ Voltage
- ✓ Current
- ✓ Position
- ✓ Temperature

Smart Servos Deserve Smart Brackets!

- ✓ *Brushed, Anodized, Aluminum*
- ✓ *Integrated Pem Nuts for Easy Construction*



**AX Side
Bracket**

**AX Short
Bracket**

**AX 45 Degree
Bracket**

**AX Foot/
Hand Set**

See our web site for pictures, videos and more!
www.CrustCrawler.com

Or call us at:
480-577-5557

NEW PRODUCTS

DEVELOPMENT TOOLS

Tools for Motor Control Designs



Microchip Technology, Inc., has announced two new low-cost development systems: one for the control of high-voltage motors and another for stepper motors. Along with related applications notes and free source-code software, these development tools enable rapid designs using dsPIC® digital signal controllers (DSCs).

The new dsPICDEM™ MCHV Development System is for the rapid evaluation and design of a wide variety of high-voltage, closed-loop motor control applications using AC induction motors (ACIMs), brushless DC (BLDC) motors, or permanent magnet synchronous motors (PMSMs). The board includes in-circuit debugging circuitry, eliminating the need for a separate debugger for development with Microchip's dsPIC33 Motor Control DSC families.

The new dsPICDEM MCSM Development Board is a cost-effective tool for creating unipolar and bipolar stepper motor applications. This board enables the rapid development of both open-loop and current-closed-loop microstepping routines using Microchip's dsPIC33 Motor Control families.

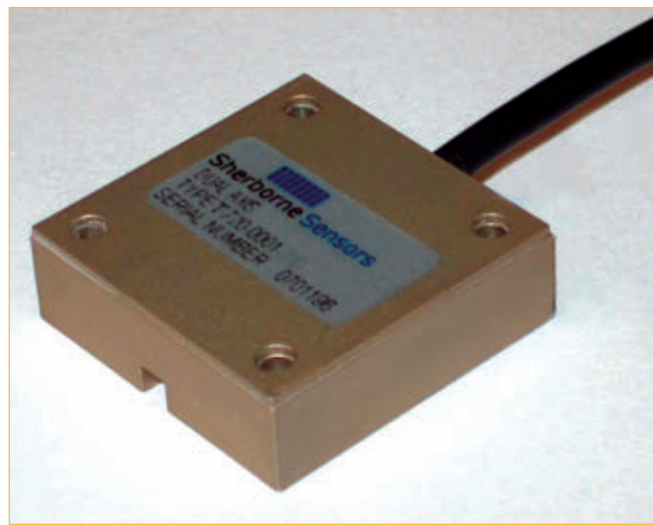
For further information, please contact:

**Microchip
Technology, Inc.**

Website: www.microchip.com

SENSORS

T700 Series Inclinometers



Sherborne Sensors has announced the launch of the T700 series: a high-precision range of compact, DC-operated, solid-state MEMS inclinometers. The T700 series is designed to measure inclination proportional to sine of angle in one axis (model T710), or dual axes at right angles (model T720) in applications where space is at a premium.

Available in ranges from $\pm 30^\circ$ to $\pm 90^\circ$, the Sherborne Sensors T700 series features an advanced micro-machined MEMS sensor, incorporating air damping. Unlike fluid-damped devices, air damping is essentially independent of temperature which enables the inclinometer to reliably perform over a very wide temperature range. Units are housed in a lightweight, robust light alloy package and are fully compensated for temperature effects on both sensitivity and zero. The T700 series also incorporates positive mechanical stops, conferring excellent shock resistance and offering reliable over-ranging protection. DC response capabilities of the T700 series facilitate exceptional measurement accuracy. Units are designed to operate from an unregulated DC power source, and offer a self-test feature on both axes.

Sherborne Sensors inclinometers are supplied with an integral shielded cable with built-in strain relief attached to the inclinometer housing which may also be fitted with an optional electrical connector. The T700 series -0101 model variant incorporates a temperature sensor which enables each unit to be individually characterized over its

operational temperature range. A supplied characteristic error correction equation helps further minimize measurement uncertainties due to zero offset, sensitivity error and linearity, thermal zero shift, and thermal sensitivity shift. By entering equation coefficients into the host operating software, cost-effective, high-precision inclination measurements with accuracies better than ± 1 milli radian may be achieved. The T700 series is also supplied with a comprehensive two-year warranty.

For further information, please contact:

**Sherborne
Sensors**

Tel: **877 • 486 • 1766** (US and Canada)
or **+44 (0) 870 444 0728** (worldwide)
Email: sales@sherbornesensors.com
Website: www.sherbornesensors.com

ColorPAL

The ColorPAL from Parallax, Inc., is a miniature color and light sensor which — through its RGB LED — doubles as a color generator. For sensing color, the ColorPAL uses its LED to illuminate a sample — one color component at a time — along with a broad-spectrum light-to-voltage converter to measure the light reflected back. The amount of light reflected from the sample under illumination from each red, green, and blue LED can be used to determine the sample's color.



Features:

- Detects a full range of colors, and outputs data as RGB (Red/Green/Blue) components.
- Detects broad-spectrum ambient light with sensitivity down to $44 \mu\text{W}/\text{cm}^2$ per lsb.
- Generates 24-bit color using onboard RGB LED.
- Plugs into servo headers (with optional cable) or wire breadboards.
- Single-pin interface uses a simple serial protocol to define and initiate color detection and generation.
- Color detection and generation details handled by onboard microcontroller.
- Onboard EEPROM for saving custom color detection and generation programs.

- Autorun feature permits running a pre-designated EEPROM program with only a power supply.
- Price is \$14.99.

For further information, please contact:

Parallax, Inc.

Website: www.parallax.com

KITS

Engineering for Kids



The 12th kit in the Little Labs line — Intro to Engineering — welcomes young children to the field of engineering with 25 experiments and building projects in five sections. By engaging in hands-on activities, children will learn how engineers apply their scientific and technical knowledge to design machines and devices.

Start by learning some engineering basics with experiments on levers, forces, and pulleys. Build simple devices using these basic components. Move on to explore engineering on land by building vehicles with wheels, such as a race car and a wind-up car. Next, investigate engineering with air and in the air by building a helicopter, a pinwheel, a parachute, a glider, a balloon rocket, and an air-powered carousel. Then, build a diving bell, a sailboat, and a paddle boat to experiment with engineering in water.

Finally, find engineering in your own home with experiments modeling the telephone and television.

With a 48-page guidebook, Intro to Engineering teaches engineering fundamentals with step-by-step, hands-on experiments and building projects.

For further information, please contact:

**Hobby
Engineering**

Website: www.hobbyengineering.com

EVENTS

Calendar

ROBOTS.NET

Send updates, new listings, corrections, complaints, and suggestions to: steve@ncc.com or FAX 972-404-0269

Know of any robot competitions I've missed? Is your local school or robot group planning a contest? Send an email to steve@ncc.com and tell me about it. Be sure to include the date and location of your contest. If you have a website with contest info, send along the URL as well, so we can tell everyone else about it.

For last-minute updates and changes, you can always find the most recent version of the Robot Competition FAQ at Robots.net: <http://robots.net/rcfaq.html>

— R. Steven Rainwater

FEBRUARY

13 National Electronics Museum Techno-Swap-Fest

Linthicum, MD

Autonomous robot contests, remote control vehicle combat, R2D2 builder demonstrations, steam engines, CNC machines, and other geek toys.

<http://groups.google.com/group/techno-swap-fest>

18-21 Teckkriti RoboGames *Indian Institute of Technology, Kanpur, Uttar Pradesh, India*

Events include automated car parking, robot combat, and robot ball games.

www.techkriti.org/#/competitions-robogames

22 APEC Micromouse Contest

Palm Springs, CA

Very fast, very small robots solve mazes very quickly.

www.apec-conf.org

25-28 Pragyan

National Institute of Technology, Trichy, India

This competition includes micromouse, square route line following, and slam dunk robot basketball.

www.pragyan.org

MARCH

7 CIRC Central Illinois Bot Brawl

Lakeview Museum, Peoria, IL

This competition has lots of events for both autonomous robots and remote control combat machines including Sumo, mini Sumo, line following, and line maze.

<http://circ.mtco.com>

12-13 AMD Jerry Sanders Creative Design Contest

University of Illinois at Urbana-Champaign, IL

Robots must move balloons around a field to play Tic-Tac-Toe. To make it more challenging, teams must perform specific tasks in order to obtain their balloons.

<http://dc.cen.uiuc.edu/>

13-14 METU Robotics Days

METU Cultural Congress Center, Turkey

A large conference that includes workshops, as well as a wide range of contests like line following, Sumo, mini Sumo, team mini Sumo, slalom, stair climbing, trash hunting, and robot triathlon.

<http://grou.ps/org>

20-21 Manitoba Robot Games

TecVoc High School, Winnipeg, Manitoba, Canada

This competition includes events such as Sumo, mini Sumo, line following, robot tractor pull, SuperScramble, Frequent Flyer, COGMATION, and Robo-Critters.

www.scmb.mb.ca

20-21 RobotChallenge

Vienna, Austria

Events included in this competition are parallel slalom, slalom enhanced, standard Sumo, mini Sumo, and micro Sumo.

www.robotchallenge.at

20-21 Roboticon

University of Guelph, Ontario, Canada

Robotic soccer for LEGO robots.

www.collegeroyal.uoguelph.ca

APRIL

8-10 BotsIQ (previously known as BattleBots IQ)

Mare Island, Vallejo, CA

In this competition, students build remote controlled combat vehicles.

www.botsiq.org

10-11 Trinity College Fire Fighting Home Robot Contest

Trinity College, Hartford, CT

Autonomous robots must navigate a scale model of a house, search for a fire, and extinguish it.

www.trincoll.edu/events/robot

13-15 DTU RoboCup

Technical University of Denmark, Copenhagen, Denmark

Varied course including line and wall following.

www.robocup.dtu.dk

15-17 FIRST Robotics Competition

Georgia Dome, Atlanta, GA

This is the big championship for FRC teams.

www.usfirst.org

15-17 National Robotics Challenge

Marion, OH

This is the NRC Championship.

www.nationalroboticschallenge.org

16 Carnegie Mellon Mobot Races

CMU, Pittsburgh, PA

This is the famous annual CMU autonomous mobile robot race.

www.cs.cmu.edu/~mobot

17 SparkFun Autonomous Vehicle Contest

Boulder, CO

Autonomous ground and air robots must circumnavigate the SparkFun building.

http://sparkfun.com/commerce/product_info.php?products_id=9016

17 UC Davis Picnic Day Micromouse Contest

University of California, Davis Campus, CA

The annual micromouse maze solving contest.

www.ece.ucdavis.edu/umouse

24 Penn State Abington Fire-Fighting Robot Contest

Penn State Abington, Abington, PA

Autonomous robots must navigate a scale model of a house, search for a fire, and extinguish it.

www.ecsel.psu.edu/~avanzato/robots/contests

24 Penn State Abington Mini Grand Challenge

Penn State Abington, Abington, PA

Autonomous mobile robots navigate an outdoor course.

www.ecsel.psu.edu/~avanzato/robots/contests

24 The Tech Museum of Innovation's Annual Tech Challenge

San Jose, CA

This year's competition is called Space Junk.

<http://techchallenge.thetech.org>

24-25 Trenton Computer Festival Robotics Contest

The College of New Jersey Campus Ewing Township, New Jersey

In addition to computer and ham radio events, there are usually several autonomous robot contests. Previous contests included maze navigation, precipice avoidance, and line following.

www.tcf-nj.org

Recycling & Remarketing High Technology

WEIRDSTUFF®

WAREHOUSE

Software, Computers, Electronics, Equipment, Doo-hickies

384 W. Caribbean Dr. Sunnyvale, CA 94089

Mon-Sat: 9:30-6:00 Sun: 11:00-5:00
(408)743-5650 Store x324

 **WE BUY AND SELL EXCESS & OBSOLETE INVENTORIES!**

FREE COMPUTER RECYCLING
We recycle computers, monitors, and electronic equipment. M-Sat 9:30-4:00 

 **WEEKLY SEALED BID SALE AUCTION**
Hi-tech items, electronics test equipment, and more!

GIANT AS-IS SECTION
10,000 sq. ft. of computers, electronics, software, doo-hickies, cables, and more! 

also check out our...

eBay Store
stores.ebay.com/WeirdStuff-Inc

WWW.WEIRDSTUFF.COM

Robotics Showcase

ALL ELECTRONICS CORPORATION

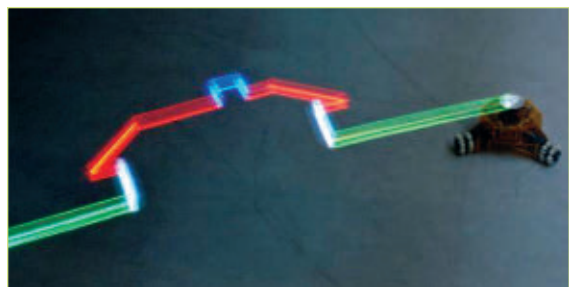
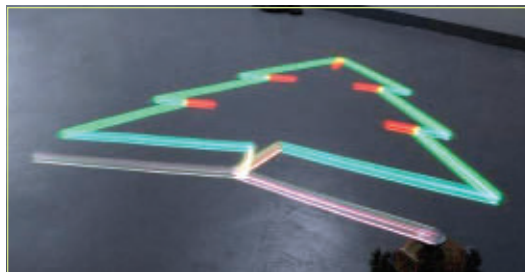
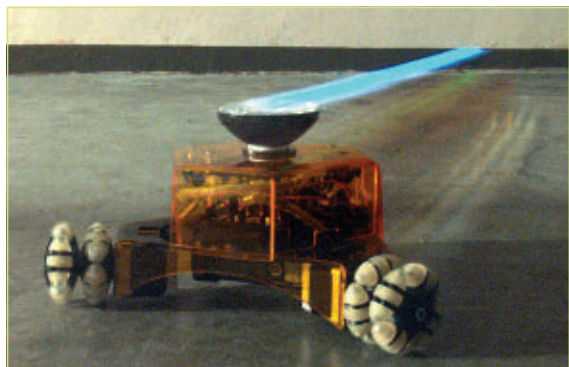
THOUSANDS OF ELECTRONIC PARTS AND SUPPLIES

VISIT OUR ONLINE STORE AT www.allelectronics.com

WALL TRANSFORMERS, ALARMS, FUSES, CABLE TIES, RELAYS, OPTO ELECTRONICS, KNOBS, VIDEO ACCESSORIES, SIRENS, SOLDER ACCESSORIES, MOTORS, DIODES, HEAT SINKS, CAPACITORS, CHOKES, TOOLS, FASTENERS, TERMINAL STRIPS, CRIMP CONNECTORS, L.E.D.S., DISPLAYS, FANS, BREAD-BOARDS, RESISTORS, SOLAR CELLS, BUZZERS, BATTERIES, MAGNETS, CAMERAS, DC-DC CONVERTERS, HEADPHONES, LAMPS, PANEL METERS, SWITCHES, SPEAKERS, PELTIER DEVICES, and much more....

ORDER TOLL FREE
1-800-826-5432
Ask for our FREE 96 page catalog

bots IN BRIEF



GO TO THE LIGHT ...

Nils Volker has been busy developing this light drawing robot. The principle behind it is simple: moving lights + long camera exposure = cool. Very cool, actually. Go to www.nilsvoelker.com/content/lightdraw/index.html for more info.

BOT KNOWS WHAT'S IN THE (GROCERY) BAG

Remember Robovie from a couple years ago, and its creepy eyes? Well, now it has eyeballs and it's able to help with your grocery shopping.

There's actually a lot going on here. Robovie is part of an integrated system that starts at home and involves other robots and devices. Specifically, the system is designed for the elderly, and includes health monitoring equipment. At home, you can make a shopping list with the assistance of ApriPoko, who might recommend dishes based on your health. The list is stored on your mobile device and when you get to the supermarket, a Robovie will help you shop for what you need — autonomously, following you with a shopping basket, reminding you what to buy and offering suggestions. Robovie can tell when you put



Robovie



ApriPoko

something in its basket and the robot will also make a little bit of conversation.

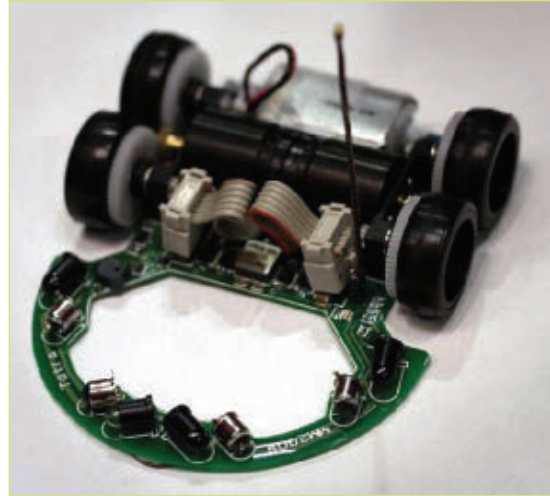
This experiment is part of a six month trial to determine how effectively a network of different robots can cooperate to assist the elderly with independent living, something that Japan specifically is very focused on due to their aging population.



bots IN BRIEF

FASTEST MICROMOUSE IN THE WEST?

Last year, the fastest recorded micromouse made it through a maze in 6.4 seconds. This year, they've gotten even faster. Apparently, there was some question as to whether a micromouse could break the five second barrier when it comes to getting through an expert class maze. Although it didn't pull it off in an official competition, a mouse called Tetra managed to make it through the maze in a seriously scant 4.7 seconds in a recent demonstration.



SHADY BOT BUILD



Daniela Rus' laboratory at MIT is in a Frank Gehry-designed building that's covered with giant windows and shiny metal — which is a bad combination when it's sunny out and you're trying to see your computer monitor. Custom window shades would have been too expensive, so instead, members of the lab built a robot called Shady.

Directed via Bluetooth, Shady autonomously climbs up the truss system supporting the windows and unfolds a shade to block just that little spot of sun that it needs to. Researchers suggest that robots using Shady's movement technique might be good for clambering around construction scaffolding or power line towers, and they've started simulating MultiShady: a combination of cooperating Shady bots with passive bars.

WHAT THE HEX?

This "mech-inspired" hexapod comes from Trossen Robotics community member Zenta. One of the cool things about this hexapod is that each of its legs has four degrees of freedom.

Get all the juicy details at: <http://forums.trossenrobotics.com/showthread.php?t=3712>.



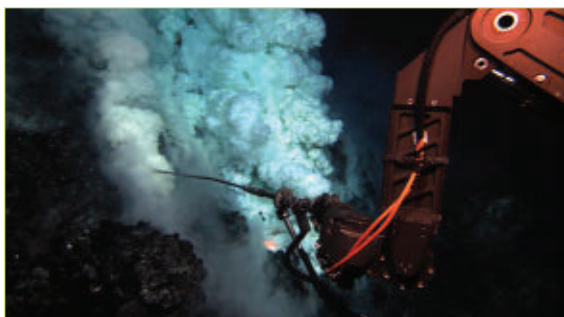
Cool tidbits and interesting info herein mainly provided by Evan Ackerman at www.botjunkie.com, but also www.robotsnob.com

LEARNS FROM ACTIONS

The ability to teach robots new tasks is rather important. But even more important is teaching robots to generalize one thing or task and apply it to many different variations on that thing or task. Recall the ASIMO that learned to generalize different types of objects, for example, show it a chair and ASIMO could figure out what other objects were chair-like.

PR2 from Willow Garage has taken this concept of generalization a step further by applying it to actions, not just things. When you teach PR2 an action (by providing an example of the action), it can understand the fundamental components of the action and apply them to different variations of the action. So, if you teach PR2 to pour a beverage out of a can into a cup, it doesn't matter what kind of can or what kind of cup, and more importantly, where they are relative to each other or if you move the can and/or cup around in the middle. This is huge for robots to be able to do, because having to teach a robot an action for "pour from can into cup on counter" and then a separate action for "pour from can into cup on low table" isn't just tedious, it's one of those things that is going to bring development to a screeching halt.

There are a lot of major obstacles to surmount when it comes to creating a robot that can be practical *and* useful around people who know nothing about robots ... from stuff like plugging into a normal outlet to recharge, to not squeezing things too hard.



HOT SHOTS

Scientists from the Woods Hole Oceanographic Institution managed to capture a volcano in action in the Pacific, about 4,000 ft. below the surface with the help from Jason — a remote controlled underwater robot. Photos and videos show that what is spewing out of the West Mata volcano is boninite lava which is thought to have only been present on extinct volcanoes. Interestingly enough, the team managed to find some shrimp in the same area and will be studying their DNA.

Jason, designed and operated by the Woods Hole Oceanographic Institution utilized a prototype, high-definition still and video camera to capture the powerful event in an area bounded by Fiji, Tonga, and Samoa.

Images courtesy of NSF, NOAA, and WHOI Advanced Imaging and Visualization Lab.

NANOS VS. HAMSTERS

Who needs a Zhu Zhu pet? When the original Hexbug Nanos were conceived, there were over 150 variations of legs alone tested, so you know that this robotic insect has a history. The final version allows the legs to bend and unbend, allowing it to move forward. When you order the color may vary, but you will get one that can not only flip over by itself but comes with an access code that allows you to go online and play games.

Oddly enough, your Nano can have its own Habitat soon, leaving us to believe that the company really is after the robotic hamster market.



ROBOT BREWING

Primula Products has a new coffeemaker — the Speak 'n Brew — that will ask you the time to set its clock and what time you would like it to brew. The 10 cup Speak n' Brew has a chip inside that understands your speech to set the clock or to program the brew time. It also works manually instead of using the Talk feature.

There are just two Simple Steps:

1) Press Talk; and 2) Listen and Speak.

Speak n' Brew answers confirming the time and shows in the display.

www.primulaproducts.com



WALL-E MAKES TRANSFORMATION

Every once in a while, WALL-E pops up in a new form. This time he takes on the Transformers. The 5.8 x 9.5 x 8.5" robotic dude starts out as a cube, but when you press buttons, he will reveal wheels, arms, and his head. A Transformer Eve is also available. She mostly resembles an egg until you press her buttons. (No sexist jokes here, please.)

COMBAT ZONE

Featured This Month:

Features

24 **MANUFACTURING:**
*Hobby Guide to Metal
Lathe Work: Part 1*
by Chris Williamson

27 **PARTS IS PARTS:**
Hole Drilling Redux
by Nick Martin

Events

27 *Nov/Dec 2009 Results and
Feb/Mar 2010 Upcoming
Events*

28 **EVENT REPORT:**
Mayhem in Miami
by Joey Daoud

ROBOT PROFILE – Top Ranked Robot This Month:

30 *3pd* by Kevin Berry

MANUFACTURING

Hobby Guide to Metal Lathe Work: Part 1

● by Chris Williamson

First and foremost is lathe safety. You are working around very fast rotational parts with high torque. Please be aware of your surroundings, and remove all jewelry, long sleeves, and anything hanging that could stand a chance to get caught up in rotational parts. If you have long hair, tie it up. Also wear safety glasses at all times to protect your eyes from chips and oil. You and ONLY you are responsible for your safety while running machinery. Always check your chuck to make sure material is tight, and get into the habit of removing the chuck key IMMEDIATELY after use, as this can become a projectile if the lathe is turned on with the key left in the chuck.

There are five basic parts to a metal working lathe. We'll go over each of these parts and explain exactly what they are for. Follow along using **Figure 1** as a reference.

The first main component to any lathe is the headstock (section A). This is the area in which all of the speed controls and feed gearing is housed, and the chuck is attached to. This is one of the most robust pieces on the lathe as it requires a lot of force on some materials for cutting. The more rugged this area is, the less likely to induce harmonic vibrations that will end up as 'chatter' in the workpiece.

The chuck is attached to the spindle on the headstock, and is normally hollow to allow longer material to be worked (section B). The chuck allows you to hold material. Most chucks are auto-centering, meaning that as you tighten down on your material, it automatically keeps the part centered. There are multiple chucks that will fit most lathes, ranging from three jaw, four jaw, and six jaw applications. Three and six jaw chucks are used for

round material, while four jaw chucks are used for square and rectangular material. Four jaw chucks take a bit more know-how to center your material, and are not recommended for beginners.

The lathe bed is the next important feature. This allows a good working base for the headstock and tailstock to be perfectly aligned, and normally will have hardened and ground slides for the carriage and tailstock to slide on. The bed carries everything for the lathe, and again should be very robust to minimize chatter so as not to reduce work quality. You will also find a rack and pinion gearing system mounted to the bed; this is how the carriage is driven.

Since you've already seen the word carriage appear a few times here, this is the next important piece (section G). The carriage is the whole cutting section of the lathe. It allows the operator to control a tool bit for facing (moving perpendicular across the material) or turning (moving parallel with the material).

There are a multitude of tool holders on the market. Most lathes from the manufacturer are equipped with a simple four post tool holder (section D), but it is highly recommended to upgrade to a 'quick change' tool holder. This will allow you to set up multiple tools and quickly change them out without having to take the individual tool itself out every time. The carriage is moved by two handwheels manually or by engaging an automatic feed system which allows the carriage to become power-assisted and give smoother finishes.

The amount of movement per one revolution of the handle depends on the lathe manufacturer. One handle (normally larger) will move the carriage itself, while the secondary (smaller) handle will control the cross slide. The cross slide is used for facing material and changing the depth of the cut. The

tool post is attached to a compound rest. It provides a smaller amount of movement along its axis (which can be adjusted on multiple angles). The compound rest is used when turning tapers or threads.

The fourth part of a lathe is the feedscrew (section F). The feedscrew is the main drive of the lathe for carriage operation. The feedscrew is driven by gears located inside the headstock and is normally able to be quickly changed by turning a handle located on the headstock itself. It allows you to change a ratio for turning and facing, as well as setting up to be able to turn threads in material.

Last but not least is the tail stock (section E). The tailstock will taper similar to the headstock to allow a multitude of tools to be mounted in line with the spindle axis. It will have a hand wheel, locking mechanism, and spindle. The spindle is set up to handle drill chucks, drill bits, centers, and other tooling. With the locking mechanism, it can be positioned anywhere on the bed to allow full

travel of the tailstock spindle. This is very useful for keeping longer material stable while trying to machine it.

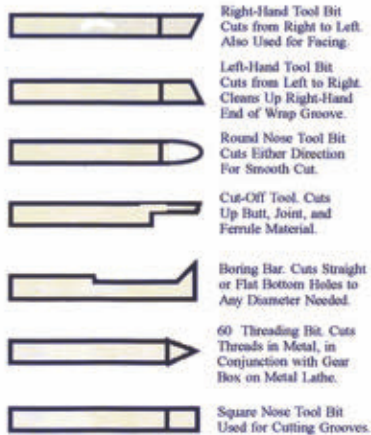
Now that we have a brief understanding of the parts of the lathe, we can get to understanding how the lathe works. On the headstock, you will probably find a lot of handles. Don't get discouraged!! These handles give you complete control over the lathe itself. Some will determine spindle rotation speed, while others will control the auto-feed functions of the carriage for turning and facing. There will also be a thread guide on the lathe itself to allow you to set the handles in the correct configuration for turning threads into material.

You should see three handles on the carriage itself. Each of these handles should have some type of dimensional guide around them. This will give you a good estimate of how much one revolution of a turn of each handle takes in material. When running a lathe, you will need a few additional

FIGURE 1.



Tool Bits Names & Functions



Courtesy of www.cuesmith.com.

pieces of tooling as well, including micrometers, calipers, and indicators to be able to measure and make sure your work piece is true and correct.

Once you have everything set up and situated, you will need to grind some tool bits or buy a set of turning and facing tools. This will allow you to begin cutting material. You will want to have the tool bit centered with the work piece. The easiest way to do this is to set a center in the tailstock and align the tool bit tip with the center. This is where the quick change tool post comes in handy as it is extremely easy to adjust the entire tool holder up and down.

You are now ready to chuck material. I recommend starting with 6061 aluminum as it is a very versatile material and easy to begin learning with, as well as being a cheap resource.

For starters, I recommend picking up a copy of *Machinist's Handbook*. This will give you a good idea for speeds and feeds, as well as tons of other information. Once you start playing with speeds and feeds, you will quickly understand that movement of the carriage is relative to the rotational speed of the work piece. The softer the material, the faster you will want to cut as to not gum up the tool

Coming Up ...

Strap in and get ready for Part 2 coming soon. In the Hobby Guide to the Metal Lathe Part 2, we will cover more machining capabilities of the lathe, going from threading, boring, reaming, and polishing. We will explain the Steady Rest (section C), and show pictures of some of the operations in this article. We'll also go over locations and costs to buy lathes so you can have a mini-lathe in your home shop or garage too!

bits. However, for starting out, slower is better as to not ruin material or tool bits.

You will learn feeds very quickly once you know what to look for in a chip. Most rotational speeds will be a few hundred rpm. I only recommend going above 400-500 rpm for polishing purposes.

Chuck your material close to the chuck while facing. If possible, this will give you a truer center as well, if you need to hang the part out for turning. Now that you have material chucked and ready to go, you will need to select a tool. A great hint for working metal or plastic is to grind your tools with a slight radius on the cutting tip, as this will provide you with a smoother surface.

A tool with a sharp pointed tip will cut grooves into the material. Tool selection is an important process to gaining great quality parts. When facing off material, if your tool is too high or too low, you will be left with a bump in the center of the part. You can use this to help adjust your tool height; once you are correct, this bump will disappear.

Again, start with a slower rotational speed and check the levers on the headstock. Set these to a very low feed. Move the tool close to the part and begin your cut using the auto feed, taking only a small amount of material. You want to take the very minimum off to clean up the edge of the part

without having a high amount of waste. In aluminum — once you figure out the speeds correctly — you should end up with a nice rolling chip. With the correct grind in the tool, you should get a broken chip while cutting. We now have a clean face and are ready to drill a center.

Get your drill chuck for the tailstock and lock it into place; put a small center drill into the chuck. Slide the tailstock up to within a couple inches of the material face and lock it into place. Again, start with a slow rotational speed. Later, you will learn speeds as you go. You will now be able to use the tailstock handle to drill into the workpiece; just deep enough to allow you to use a center to hold the part.

Once complete, back the tailstock off and change the drill chuck to a live center. Unchuck your material and slide it out until the desired length of turning is reached; retighten the chuck. Now with the center in the tailstock, slide it up and lock it into place where the center is touching the material. This helps center and stabilize the material for turning.

Change your tools to a turning tool. Again, a slight radius on the tip will work wonders for the finish; you may choose to take a very small 0.010" to 0.020" finish pass once you have your other turning complete. Make sure the tool is centered on the part and change your tumbler gear to allow the auto feed to engage the turning gear on the carriage. Set your tool to barely touch the material and turn on the auto feed. You will be after a nice broken chip here, as well. If you start getting vibrations or see ridges forming in the part, back the tool off and try a smaller cut.

Sometimes taking your time will make a lot of difference in finish quality. This is also handy to take a finish pass to clean up the material with a final cut. **SV**

PARTS IS PARTS: Hole Drilling Redux

● by Nick Martin

Back in the November '09 issue, I ended the Hole Drilling story with a reference to sharpening drills. I have now purchased a Drill Doctor 750 sharpener and the results are surprising and worth sharing. The sharpener comes in a very solid plastic case that should be a lesson to all other power tool manufacturers.

The sharpener has a plastic body and a mix of plastic and metal fittings. While I was put off by all the plastic at first, the design seems strong enough for years of use in a home workshop.

Remarkably, the sharpener paid for itself in just three weeks. With street prices under \$150 USD and the high price of large drills, you only have to salvage 10 to 20 old drills over 1/2" to break even. While the 750X model takes up to 3/4" drills, you should be aware that

larger drills take much longer to sharpen.

All the Drill Doctor tools come with a detailed manual and DVD that will get you to expert level in no time. Here are a few extra tips that are worth knowing:

- Getting the coarser grit wheel really cuts down the time to sharpen larger drills. It's well worth the expense.
- The dust produced during operation is fine, magnetic, and probably conductive. Don't sharpen drills anywhere near electronics or motors.
- Some brands of drills take split points better than others. Try the feature out and don't be surprised if you have to grind off a huge amount of steel.
- Changing the point angle takes a long time. I found it quicker to



Drill Doctor and re-sharpened drills.

roughly shape the bit on a bench grinder, then finish it on the Drill Doctor.

A tip for international readers: The Drill Doctor tools often carry a high import premium outside the US. I saved about \$100 by buying the 750X mail-order from the US and running it from a 220V to 110V step-down transformer. You can run the tool from any mains voltage as long as there is an appropriate step-down transformer available. The motor in the Drill Doctor is actually a DC type so mains frequency shouldn't matter. **SV**

EVENTS

Completed and Upcoming Events

Completed Events for Nov 11 to Dec 7, 2009

Mayhem in Miami was presented by the 305 Robotics Consortium, Starbot, Ransom Everglades, and the Symbiosis Foundation on November 14th.

Upcoming Events for February-March

NEW Model Hobby Expo 2010 will be presented by Western



Allied Robotics in Monroe, WA, on February 13th. Go to www.westernalliedrobotics.com for more information.

Motorama 2010 will be presented by the North East Robotics Club in Harrisburg, PA, February 19th through 21st. Go to www.nerc.us for more information.



Central Illinois Bot Brawl 2010 will be presented by Central Illinois

Robotics Club in Peoria, IL, on March 6th. Go to <http://circ.mtco.com> for more information.



2010 Robot Rumble will be presented by Carolina Combat Robots in Durham, NC, on March 20th. Go to www.carolinacombat.com for more information. **SV**



EVENT REPORT: Mayhem in Miami

● by Joey Daoud

At Mayhem in Miami, it wasn't without a bang that Fluffy earned the name Flamin' Fluffy, and made the South Florida robotics event be true to its name.

Mayhem in Miami was held at Ransom Everglades on November 14. It was co-hosted by the 305 Robotics Consortium, Starbot, Ransom Everglades, and the Symbiosis Foundation. All the participating teams were from schools in Miami (Carrollton, Columbus, Ransom Everglades, Miami Northwestern, and Miami Lakes Educational Center).

There were two events: 15 lb. BattleBots and a task competition.

The task competition was added due to its wild popularity with students and family at the Miami Science Museum competition held a few weeks earlier. Twenty balloons are put in a square arena; half red, half blue, each worth one point. Tied to the back of each robot is a white balloon worth two points. Each team must pop an assigned color balloon and the white balloon on the other team's robot, while at the same time protecting their own.

Robot popping designs ranged

from sharpened metal, paper clips, and razor blades, to syringes and metal brushes.

In the end, Columbus took all three places. Forklift, which came in first, had a row of syringes that could be raised or lowered based on the balloon height. Since their first place win at the Miami Science Museum competition, they added a side arm with a syringe that allowed them to continue to pop balloons, even when their opponent was engaging them from the front. Forklift also won Coolest Design for task robots.

In second place was Schwinger — a robot built completely by freshmen — beating out the senior robot R-1 which came in third. However, R-1 did receive the Best Engineering award for task robot.

For the 15 lb. BattleBots competition, there was a strong showing from Ransom Everglades and Miami Lakes Educational Center, with additional robots from Carrollton and Columbus.

Some notable fights were Fluffy (Ransom Everglades) — a rectangular robot with angled sides and a six inch drum with two teeth opposite each other — against

Robokill (Carrollton) — a low profile invertible bot with angled ends, similar to a rubber eraser.

Let's just say Robokill was not the one doing the killing. Within the first few seconds of the match, Fluffy's drum flipped Robokill across the arena. In doing so, Fluffy was also flipped upside down. However, thanks to some newly added side attachments, Fluffy was still able to drive and eventually flip itself back over.

With a righted Fluffy, Robokill was tossed around the arena until Fluffy ripped its top off, ending the match in a knock-out.

Another robot everyone was eager to see finally work — or more how it worked — was ROFL, designed by the same team behind Fluffy (for those unfamiliar with text-speak that stands for Rolling on the Floor Laughing, and you'll understand why in a second). ROFL is basically a cylinder wrapped in rubber with two tapered arms on the side attached to motors. When the motors spin, the arms push down and allow the cylinder to roll forward in a flailing, wobbling fashion. While ROFL has no weapon, its defense is great in that it can just roll over its opponents.

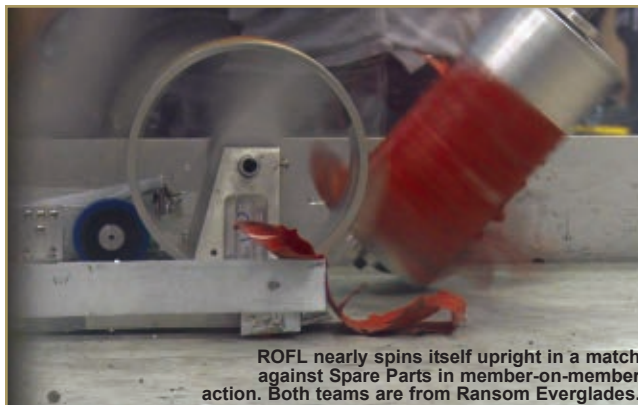


Miami Lakes vs. Ransom shows why sparks are such a crowd pleaser!



Fluffy shows what happens when magic smoke turns into magic fire.

Bill Garcia removes Fluffy after the flames have been extinguished. Photo by Andrew Hevia.



ROFL nearly spins itself upright in a match against Spare Parts in member-on-member action. Both teams are from Ransom Everglades.

With its arms flailing and making loud taps on the metal floor, many described it as an angry robot. The team eventually got the hang of the controls so well that they were able to create a gyro effect with ROFL and nearly wobble it upright. It even won a match against Spare Parts, another Ransom Everglades robot. Fluffy received Best Engineering for 15 lb. robots, and ROFL received Coolest Robot.

Then came Fluffy versus TTATT, both undefeated. TTATT is a low profile wedge robot from Miami Lakes Educational Center. Its wedge is on a hinge, so it can tilt up and down.

As soon as the match began, both robots charged head on, with TTATT's wedge extended outright, parallel to the floor. Fluffy made contact and both robots shot in opposite directions, landing in different corners of the arena. Fluffy was turned upside down again.

The robots scurried around the arena for a bit, and then the flames started. At first it was a quick flash of a small fire, but then it grew bigger and constant, with the flames almost reaching two feet high by the time Bill Garcia opened the arena and the fire was extinguished by one of the maintenance workers. Everyone was cheering. Even Fluffy's team thought it was the greatest thing ever. "Bet you're all glad you stayed now," remarked the announcer. Just as everyone thought the excitement was over, Fluffy erupted into flames again, prompting another spray of

fire extinguishers.

By then, the fumes from the extinguishers became overwhelming, and everyone was ordered to evacuate. All the parents and students shuffled out of the second floor gym to the patio area below.

"When we hit TTATT, the drum shifted sideways so much that the pulleys became misaligned," explained Will Bales afterwards, one of Fluffy's builders. "When that happened, the belts jammed, making the motor screech to a halt. That made several MOSFETs in the speed controller burst into flames. That overloaded the battery, which eventually caught on fire, as well. All the rest of the electronics fell victim to the flames."

One of the students put it best.

"Fluffy wins with style and Fluffy loses with style."

Nola Garcia, the head of BattleBotsIQ and Starbot, got the excited students attention. "Talk about Mayhem in Miami. Who says math, science, and engineering aren't fun?" Everyone cheered.

It was decided, and to no objection, that the undefeated TTATT would be awarded first place. Tied for second was Fluffy, R-6, and Rattler.

The 305 Robotics Consortium is selecting a date for a spring fight sometime in February.

You can view videos from the match on Facebook at the 305 Robotics Consortium Page or the Bots High Page — a documentary is in production on BattleBotsIQ. **SV**



**MOTORS
GEARBOXES
WHEELS
AND MORE**

BB BaneBots

**BANEBOTS.COM
970-461-8880**



ROBOT PR FILE

TOP RANKED ROBOT THIS MONTH

● by Kevin Berry



3pd

Class: 3 lb Beetleweight
Team: Ti Joe
Builder(s): Andrew Peterson
Location: State College, PA

<u>BotRank Data</u>	<u>Total Fights</u>
Lifetime History	69
Events	17

<u>Wins</u>	<u>Losses</u>
48	21

3pd has competed in: Robot Club & Grill (RCG)-301; Motorama 2003; RCG-303; RCG-304; RCG-305; 12 Lbs of Pain; Robot Assault 2003; House of NERC 2003; Motorama 2004; Pound of Pain 7; Robot Assault 2004; House of NERC 04; Motorama 2005; Robotic Revolution — New Orleans; House of NERC 05; 2005 RFL Nationals; and Motorama 2006. Details are:

● **History:** 3pd was ranked #1 in April 2003, and then for 22 of the 26 months between July 2005 and August 2007. It also held 2nd or 3rd ranking 23 other months. This awesome record of 46 months in the top three earns it a place in Combat Robot History as a Top Bot.

● **Configuration:** Two wheel drive vertical spinner.

● **Frame:** 0.22" 6AL-4V titanium, 0.375" 7075-T6 aluminum.

● **Drive motor:** 33:1 Escaps with outboard nylon bearings.

● **Wheels:** 3" x .75" Colsons.

● **Hubs:** 0.75" keyed UHMW.

● **Drive ESC:** Custom esc.

● **Batteries:** 14.8V x 1,300 mAh LiPoly.

● **Weapon:** S7 tool steel, asymmetric counterbalanced tooth design beater.

● **Weapon motor:** JETI 15-4Y brushless motor.

● **Weapon controller:** Castle Creations Phoenix-40 ESC.

● **Armor:** 0.02" 6AL-4V titanium.

● **Radio system:** Spektrum DX6.

● **Design philosophy:** One hit wonder.

● **Builder's bragging opportunity:** Combat record is 73 wins, 24 losses. **SV**

Information and photo credits courtesy of Andrew Peterson. All fight statistics are courtesy of BotRank (www.botrank.com) as of December 6, 2009. Event attendance data is courtesy of BotRank and The Builder's Database (www.buildersdb.com) as of December 6, 2009.



1st Annual Techno-Swap-Fest

Feb 13, 2010 - 9am to 2pm
National Electronics Museum
1745 West Nursery Road, Linthicum MD



Attention All Hobbyists

- *Is your workshop so cluttered that you can barely work on your projects?*
- **Did you buy something online that didn't fit into your plans?**
- ***Are you still looking for that perfect gadget, part or tool?***

If you answered YES or NO to any of these questions, you need to attend the 1st Annual **Techno-Swap-Fest** to buy, sell and gawk at all the precious junk that other hobbyists crave.

Here are just some of the hobbies expected to be represented:

- R/C (boats, planes, tanks, cars, trains, geese, monsters)
- Robotics (competition, combat, R2D2, micro-bots, junk-bots)
- Electronics (hackers, makers, crackers, phreaks, HAMs)
- Mechanical (steam engines, CNC, animatronics, whimsical)

And some of the precious stuff (a.k.a. junk) you'll find:

Motors, gears, tools, parts, radios, batteries, materials, wheels, circuits, gadgets, cases, plans, complete projects, never-completed projects, models, books, and much, much more



***** Door Prizes : Most Unusual, Least Useful, Best Dust Collector *****



All proceeds support the National Electronics Museum:

Entrance Fee:	Adults \$5, Kids under 16 FREE
Seller's Table:	\$10 (limited number, reserve early)
Club Sponsorship:	FREE (just send us your info)

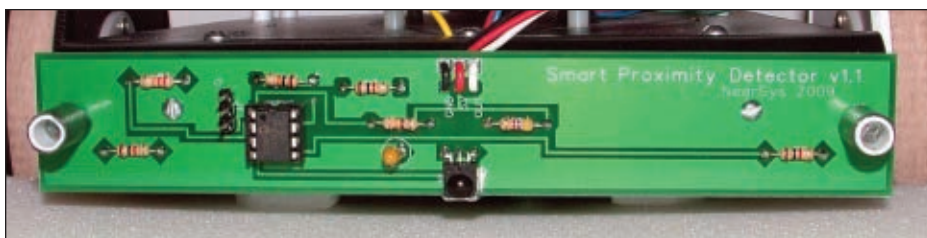
For more information or to reserve a table send email to:

techno.swap.fest@gmail.com

<http://groups.google.com/group/techno-swap-fest>

*** Your choice for a wintry Saturday morning ***

- (a) Attend Techno-Swap-Fest and experience the thrill of hunting for junk
- (b) Help put up new wallpaper in the guest bathroom



The CheapBot Smart Proximity Detector

The PICAXE-14M inside the CheapBot-14 (our small platform we've been working on) robot controller only has 256 bytes of memory. While that's fine for a basic robot, it leaves something to be desired when trying to create a more capable robot. So, why not use a second microcontroller to offload some of the PICAXE-14's work? The smart proximity detector outlined here incorporates an inexpensive PICAXE-08M that takes care of detecting obstacles and leaves the driving to the robot controller.

Not familiar with a proximity detector? A proximity detector is an infrared (IR) bumper system used to detect collisions before they become collisions. As the robot carrying the proximity detector approaches an obstacle (like a wall), the detector raises the alarm in time to avoid a collision.

The first proximity detector I read about was described in the book, *Mobile Robots*, by Jones, Seiger, and Flynn. Their detector uses an inverter, resistor, and capacitor to create a pulsed 40 kHz signal. Other designs use a 555 (a dual 555 IC) timer in place of the inverter. Anodes of two infrared LEDs (IREDS) are connected to the timer. By bringing the cathodes of the IREDS to ground one at a time, each is activated from the same 40 kHz timer. I like that idea; use a single timer for both IREDS, but only let one flash at a time.

Due to the tolerances in the timing components (the capacitor and resistor), a trimmer is required to properly set the frequency of the IREDS. So, what if a PICAXE-08M was used instead? Setting the proper frequency of the IREDS

would be as simple as writing a single line of code. However, using a PICAXE-08M to blink two IREDS replaces 50 cents in parts with a \$3 chip with lots of spare memory. Blinking the IREDS barely taps into the capability of the PICAXE-08M.

So, what about adding another feature to the proximity detector, like monitoring an IR detector? That gives the PICAXE the responsibility of monitoring reflections from the IREDS rather than the robot controller. The results can be reported to the robot controller over a serial data link.

Since *Mobile Robots* was published in 1999, there's a greater number of 40 kHz IR detectors available. So, I switched from the old Sharp sugar cube IR detector they used to a ROM 37.9 kHz receiver module (part number G16943 from Electronic Goldmine; www.goldmine-elec.com). Switching to a 38 kHz detector posed no problem; I just needed to change one line of PICAXE code.

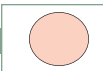
Since there's a single PWM timer available on a



The IR detector I selected for the Smart Proximity Detector. It's a three-pin device that's simple to use. When it detects an IR pulse, its output goes to ground.

PICAXE-08 (pin 5), the anodes of both IREDS connect to the PICAXE here. The cathodes of the IREDS however, connect to different PICAXE I/O pins (3 and 6). By LOWing these pins one at a time, each IRED is activated at its proper time.

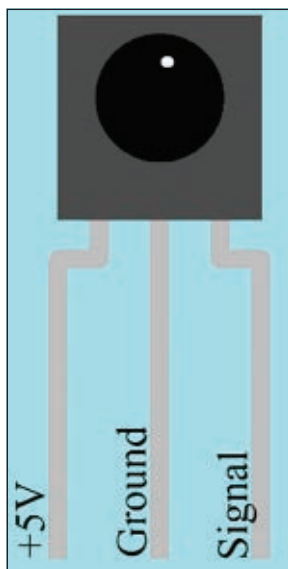
It sounds good so far. The PICAXE-08M provides the



proper 38 kHz timing, pulses each IRED, monitors the IR detector for reflections, and then reports the results to the robot controller. That still leaves plenty of memory left over. What other functions are available for the proximity detector?

At a Boise Robotics Group meeting several years ago, Barry Nye, the Technology Guy, demonstrated a BOE-Bot estimating the distance to an obstacle by changing the pulse frequency of an IRED. The PICAXE-08M can do likewise. This works because the IR detector has a bandwidth of detection frequencies. The center of the bandwidth for the ROM is 37.9 kHz. However, it's also sensitive to other frequencies — just to a lesser extent. The farther from the center of the bandwidth the IR pulse frequency is, the brighter the IR pulses must be in order for the ROM to detect them.

Since the IREDs are supplied with a constant current, their intensity is also constant. The only way their intensity gets dimmer is if the distance to the reflecting obstacle is greater. By changing the pulse frequency of the IREDs and monitoring for a reflection, the proximity detector can estimate (crudely) the distance to an obstacle.



The pinout for the ROM 37.9 kHz IR Receiver Module.

Table 1 is based on my experiments and shows the approximate distances that the ROM can detect an IR reflection from the 940 nm IREDs used in the Smart Proximity Detector. I've also included the PWM settings for the PICAXE-08M pulsing the IREDs.

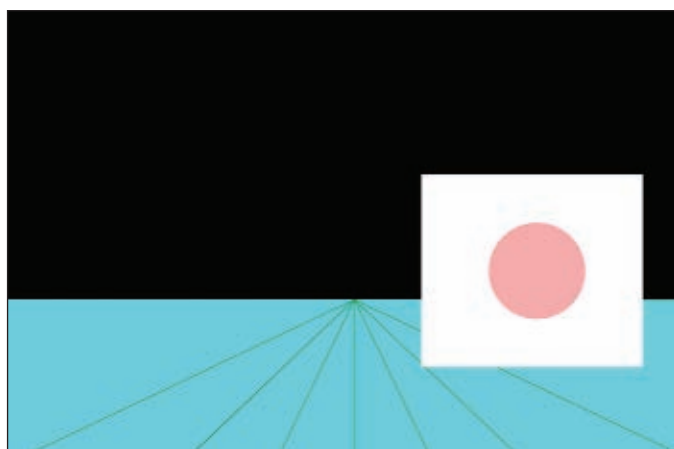
Note: The IRED supply voltage is five volts and the resistor in series with each IRED is 1K ohms. The PWM wizard in the PICAXE Editor calculated the PWM settings.

Making the Smart Proximity Detector

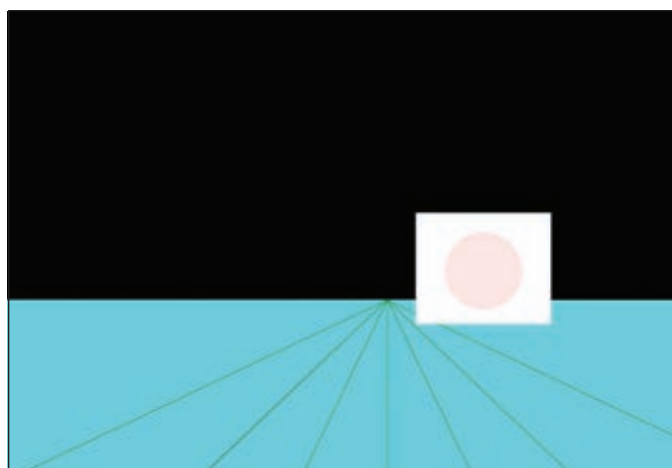
You'll need the following parts to make a Smart Proximity Detector:

- (2) IREDs (T-1-3/4 940 nm)
- (2) Plastic or aluminum tubing (1/4" diameter) *
- (2) 1K resistors (all resistors are 1/4W)
- 10K resistor
- 22K resistor
- 330 ohm resistor
- 4.7K ohm resistor
- 4.7 μ F electrolytic capacitor (five volts or higher and 0.1 inches between leads)
- (2) Three straight pin headers (0.1 inches between leads)
- Eight pin IC socket (0.3 inches wide)
- PICAXE-08M
- IR receiver module (anything similar to the ROM will work)
- Stranded wire (I use #24 AWG)
- Thin heat shrink (to cover the above wire)
- Smart proximity detector PCB (use the pattern provided at www.servomagazine.com)

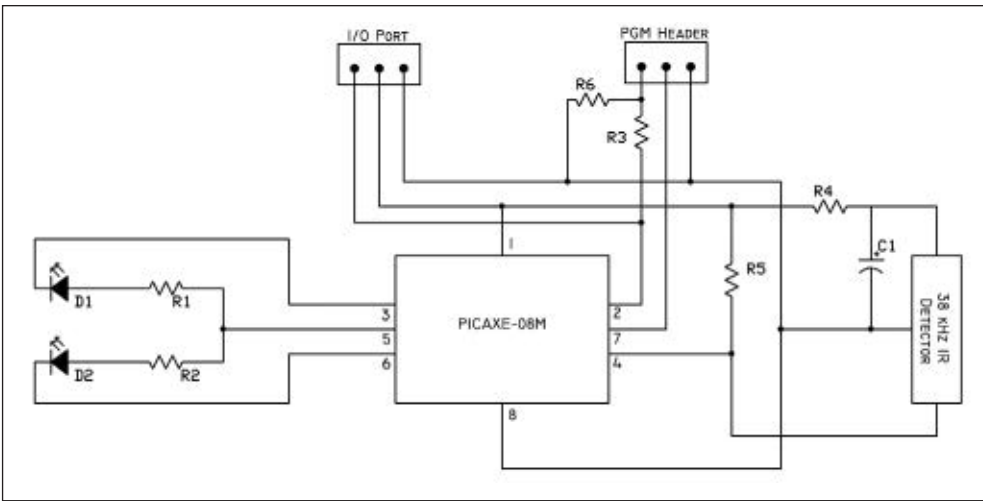
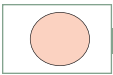
TABLE 1.		
Frequency	Distance	PWM Setting
38 kHz	14 inches	2,25,53
37 kHz	10 inches	2,26,54
36 kHz	9 inches	2,27,56
35 kHz	6 inches	2,28,57
34 kHz	5 inches	2,28,59
33 kHz	3 inches	2,29,61
32 kHz	2 inches	2,30,63



The IR reflecting off a wall this close to the proximity detector is detectable at many frequencies, including those way off the center frequency of the IR Detector. Therefore, all PWM frequency created by the PICAXE results in a detection of this obstacle and the proximity detector reports the wall is relatively close to the robot.



The IR reflecting off a more distant wall is dimmer and can only be detected if the IREDs are pulsed at close to 37.9 kHz. Since no other frequencies will produce a detection, the proximity detector reports the obstacle is relatively distant from the robot.



This is all the parts tied together to make a NearSys Smart Proximity Detector. Notice how both IRED anodes are connected to the PICAXE's PWM pin 5 but can't be activated until their cathodes are LOWed on either pins 3 or 6. The I/O port provides power to operate the Smart Proximity Detector and transmits proximity reports.

* If you use a plastic tube to cover the IREDs, you need to wrap them in a layer of aluminum duct tape.

You'll note there are six jumper wires in the PCB. Use cut resistor leads for these jumpers to reduce your waste. Of all the parts, only the IR Receiver, IREDs, and electrolytic capacitor are polarized. Technically, the IC socket can be soldered right side up or upside down. However, it's best to solder it notch up. That way, you're less likely to insert the PICAXE-08M upside down.

Start with the resistors since they lie the lowest. As you solder them, use their cut leads to make the jumpers. Then, add the socket (don't insert the PICAXE at this time), IR receiver, capacitor, and programming header (a three-pin straight header). It's best to use three differently colored wires (red for +5V, green for ground, and a third color for signal) for your Smart Proximity Detector. The color of the wires will indicate the function of each wire in the cable which is helpful when you interface the detector to your

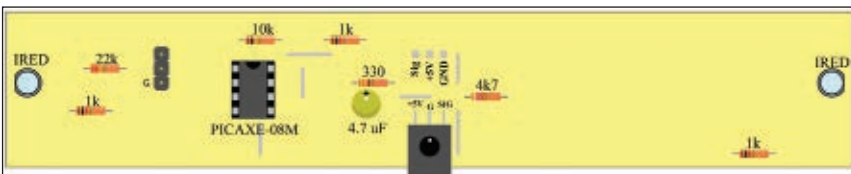
robot. Strip 1/4 of insulation from one end of each wire and solder it to the PCB. Your PCB shouldn't require strain relief once it's attached to your robot. (Although, it doesn't hurt to exercise care with the wires while you complete the detector.)

Complete the cable by terminating it with a header. I use the same method to terminate sensors as discussed in *Mobile Robots*. Tin the three short pins of the second header, as well as the free ends of the three wires (after stripping them back 1/4"). Slide 1/2 inch of heat shrink over the wires and push them up against the PCB

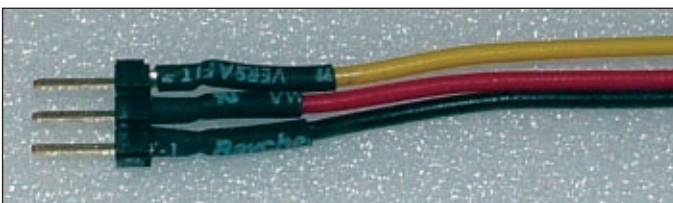
where they can remain cool while you solder. Hold a tinned wire against a tinned header pin and heat both with a well-tinned soldering iron. The solder on both the wire and header pin will fuse together making an electrical and mechanical connection. Repeat for the other two wires. After the connections have cooled, slide heat shrink over each soldered connection and shrink them.

The directions above for completing the cable assume your robot's sensor interface is like a CheapBot (the same arrangement of pins as in a servo cable). It doesn't really matter where the Smart Proximity gets +5 volts and ground, but it is simpler if the power, ground, and signal can go to the same I/O port.

To constrain the beam direction of the IREDs, you need to place collimating tubes over them. Aluminum tubes are great, but plastic ones are easier to cut and trim. However, IR is a tricky customer. What looks opaque to our eyes may be perfectly transparent to IR. So, if you decide to use 1/4" diameter plastic tubing for the collimators, wrap them in a layer of aluminum duct tape to prevent IR from leaking out. (By the way, tubes 1/2" long are adequate for the collimators.)



Parts placement for the Smart Proximity Detector.



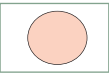
An example of a terminated cable. Because of the wire colors, I know which is signal, power, and ground before I plug it into my robot controller.

Smart Proximity Code

The code for the Smart Proximity Detector is available for download on the *SERVO* website. Feel free to modify it. At the moment, it just detects obstacles on the left and right side and reports their distances. You'll need the PICAXE Editor to program the PICAXE-08M and a three-pin to serial adapter. The software is free from the PICAXE website at www.rev-ed.co.uk/picaxe/software.htm.

The programming cable connects female DB-9 pins 2, 3, and 5 to a three-pin receptacle. The order to solder the wires to the receptacle pins is shown in the **photo**.

After programming your Smart Proximity Detector with



the sample code, it will begin producing messages with the following format.

```
Start, L, left_distance, R, right_distance
```

Start is eight high bits (a hex FF or decimal 255). After the letter L is the distance to the obstacle on the left side of the detector in a one byte field that ranges from one to seven. A seven means there was no detection of an obstacle, while a value from one to six represents increasingly greater distances to the obstacle. The one byte digit following the letter R follows the same format.

Here's a sample of the code your robot can use to monitor and respond to the distance reports from the Smart Proximity Detector. The code comes from a PICAXE-14M in a CheapBot-14 robot controller.

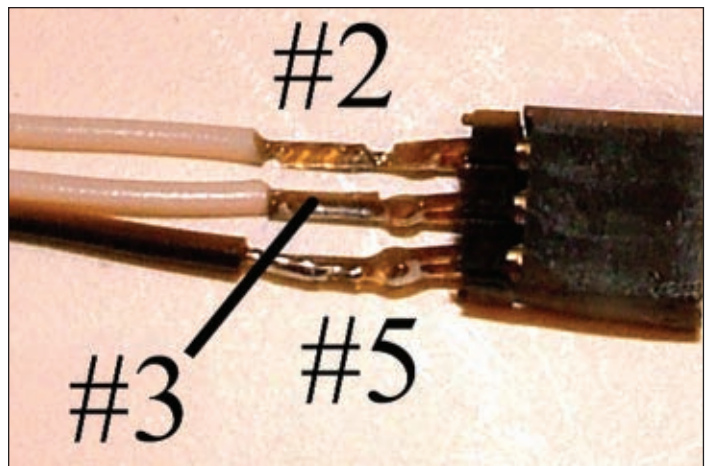
```
SYMBOL LeftDetect = B0
SYMBOL RightDetect = B1
SYMBOL Detected = B2

Traveller:
  gosub Proximity_Check
  Detected = LeftDetect + RightDetect
  if Detected > 12 then Traveller
  if LeftDetect < RightDetect then
    Turn_Right
    if RightDetect < LeftDetect then Turn_Left
    if RightDetect > 3 then Traveller
    gosub Freeze
    pause 1000
    gosub Right

Proximity_Check:
  serin
  2, T1200_4, ("L"), LeftDetect, RightDetect,
  RightDetect
  return
```

This code snippet monitors the 1200 baud messages from the Smart Proximity Detector over I/O port 2. If the sum of the left and right distances is greater than 13, then no obstacle was detected and the robot goes about on its merry way. Otherwise, it turns away from the closest obstacle unless the obstacles are equidistant on the left and right side. In that case, the robot stops and makes an about-face before continuing on.

For readers who would rather not make their own PCB, check out my website (NearSys.com/catalog). I'm selling a Smart Proximity Detector kit for \$20 plus \$4 shipping. You'll also find directions and a demonstration video along with the kit. NearSys is also selling a programming kit that's useful for three-pin programming headers.



Pin 5 is the ground wire.

End Detected

There you have it. A Proximity Detector with a little brain that offloads some of the work for your robot controller. The Detector is not limited to mobile robots, either. You can experiment with modifying its code for a stationary application that reports when obstacles are present, absent, too far, or too close. Have fun with it and let me know what you come up with. **SV**

THE ORIGINAL SINCE 1994

PCB-POOL®

Beta LAYOUT

Servicing your complete PCB prototype needs :

- Low Cost - High Quality PCB Prototypes
- Easy online Ordering
- Full DRC included
- Lead-times from 24 hrs **NEW !**
- Optional Chemical Tin finish no extra cost **NEW !**

FREE LASER STENCIL WITH ALL PROTOTYPE PCB-ORDERS

Watch "ur" PCB
Follow the production of your PCB in **REALTIME**

email : sales@pcb-pool.com
Toll Free USA : 1 877 390 8541
www.pcb-pool.com

Beta
LAYOUT

The μOLED-128

This month, we're going to once again orbit on the fringes of robotic science and wander into mechatronic space.

If you've been following the Design Cycle series over in *Nuts & Volts*, you already know that we've been working with Microchip's newest 32-bit PIC — the PIC32MX795F512L. In recent Design Cycle discussions, we've used the PIC32MX795F512L to blink LEDs and hook up with a PC host in the guise of a HID-class device. It's time to bring the resources of the 32-bit PIC32MX795F512L to a *SERVO* project. So this time around, we're going to put the PIC32MX795F512L to work as a HID-capable graphics display module driver.

The μOLED-128-G1

The 4D Labs μOLED-128-G1 Display Module is based on Passive Matrix OLED technology (PMOLED) which is driven by a GOLDELOX graphics processor. The GOLDELOX graphics engine and the OLED display have the ability to stand alone when their resources are combined. However, the μOLED-128-G1 can also be targeted by a microcontroller

and, since we have a very powerful and easy to use PIC at our disposal, that's exactly what we're going to do.

The μOLED-128-G1 is an intelligent display module that can operate in two software-determined modes. Depending on the PmmC (Personality-module-microcontroller Code) that is loaded in the GOLDELOX, the μOLED runs as a serial command platform or a 4DGL (4D Labs Graphics Language) platform. As a 4DGL-based device, the μOLED has the ability to operate as a stand-alone device under the control of a high-level, graphics-oriented language (4DGL) that is similar in syntax to C. While operating in 4DGL mode, the GOLDELOX graphics processor has the programmable smarts to take advantage of its library functions embedded within its graphics language.

The hardware resources of the GOLDELOX graphics engine are also exposed in 4DGL mode. With the assistance of the 4DGL programming language and library functions, the μOLED-128-G1 programmer can assemble and compile GOLDELOX code to control the GOLDELOX's serial port, I/O

pins, and OLED graphic display. The μ OLED-128-G1 also supports a micro-SD card which falls under the control of the 4DGL programming language as well. The built-in micro-SD card can be used to store images, slide shows, and code.

The μ OLED tumbles out of its anti-static bag preloaded with the PmmC. Serial command platform mode places the μ OLED-128-G1 in a slave configuration which allows an external host to drive the display module using serial commands. μ OLED serial commands fall into three groups; General Command Set, Display Specific Command Set, and Extended Command Set. The Extended Command Set allows the μ OLED programmer to access objects that reside on the micro-SD card.

A five-pin interface (VCC, TX, RX, GND, and RESET) connects the μ OLED-128-G1 to its host. The μ OLED requires VCC to lie between 3.6 and 5.0 volts. If you decide to include a micro-SD card in your design, you'll most likely need to supply the VCC pin with 5.0 volts as some micro-SD cards require at least 4.0 volts for proper operation. Natively, the GOLDELOX IC is a 3.3 volt part. Thus, the GOLDELOX TX and RX pins are based on a 3.3 volt power rail which is provided by an on-board +3.3 volt LDO voltage regulator. However — thanks to on-board current limiting and pullup resistors — the μ OLED-128-G1's RX pin is 5.0 volt tolerant. The μ OLED's active-low RESET pin is designed to be driven with an open collector arrangement. Dropping the μ OLED-128-G1's RESET pin logically low for more than 20 μ s will reset it. You'll see the front end of my μ OLED-128-G1 in due time. The rear end of my μ OLED is shining under the lens of my brand new Canon 7D in **Photo 1**.

Something Old, Something New

Most everything we'll do with the μ OLED-128-G1 will be done via a three-wire serial interface (TX, RX, and GND). Since we don't really have any direct physical control over the μ OLED embedded hardware, I didn't feel that posting a schematic of the display module would be productive. However, we do need to understand how to interface a microcontroller's serial interface with the μ OLED-128-G1 and that's where **Figure 1** comes in.

The PIC32MX795F512L supports up to six RS-232 interfaces (UART1A, UART1B, UART2A, UART2B, UART3A, and UART3B). Our design's serial interface selection is dependent on the available I/O that is pinned out on the PIC32 TRAINER under the

lights in **Photo 2**. As you can see in **Schematic 1**, I have chosen to use UART2A which shares pins with the PIC32MX795F512L's I/O ports RG7 and RG8. The μ OLED active-low open collector RESET circuit is derived from the PIC32's RG6 I/O pin which we will configure as an open drain.

There's nothing worse than getting up on your donkey and making assumptive plans for hardware yet untested. The PIC32MX795F512L's UART and the μ OLED-128-G1 can communicate serially without the aid of an RS-232 converter IC. However, I don't like riding on my donkey and I would like to test the PIC UART interface before committing it to the μ OLED. So, I've called upon a couple of old friends to help me

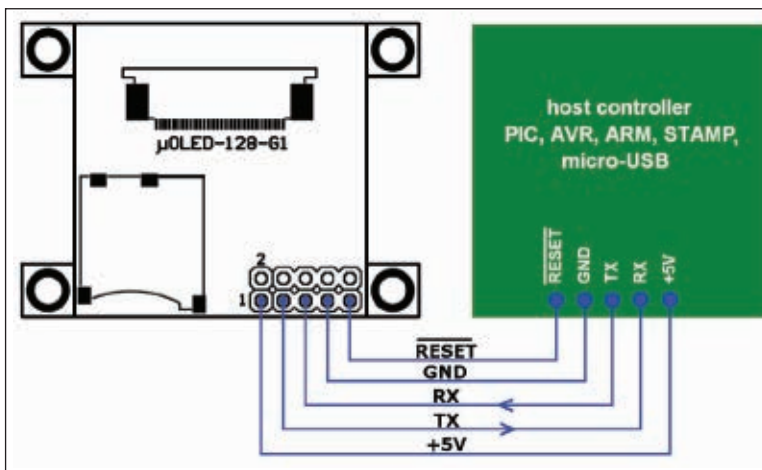


FIGURE 1. Our host controller is the Microchip PIC32MX795F512L. The PIC32MX795F512L is a 3.3 volt part and we can implement an open collector RESET circuit by simply configuring the associated PIC32 I/O pin (RG6) as open drain.

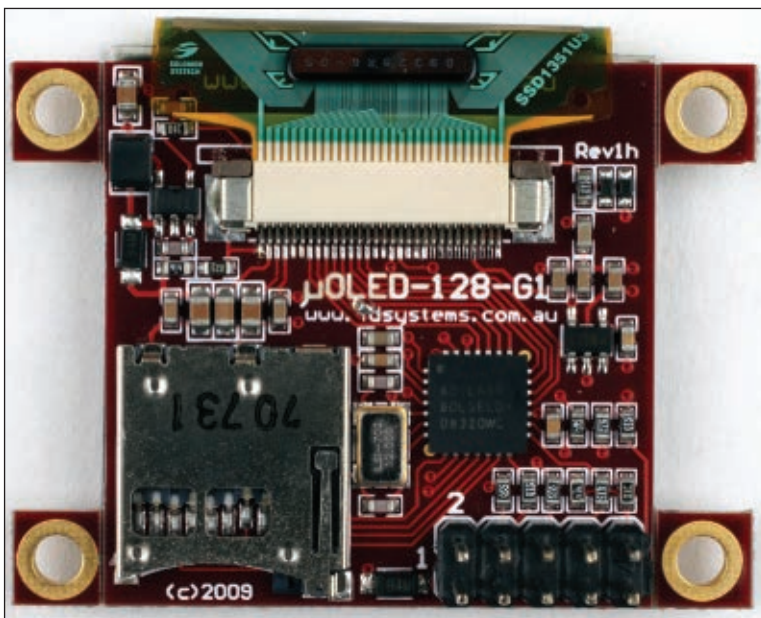


PHOTO 1. If you're interested in how the μ OLED-128-G1 is put together, the 4D Labs folks furnish a schematic in the μ OLED Users Manual. All we really need to know about the μ OLED-128-G1 hardware is how the 10-pin male connector is pinned out.

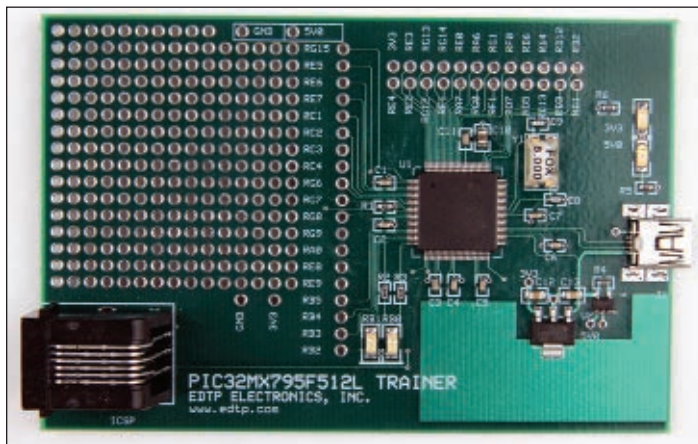


PHOTO 2. The PIC32MX795F512L TRAINER is a product of the *Nuts & Volts* Design Cycle column. With some very minor modifications, the TRAINER is a perfect host for our μ OLED display module.

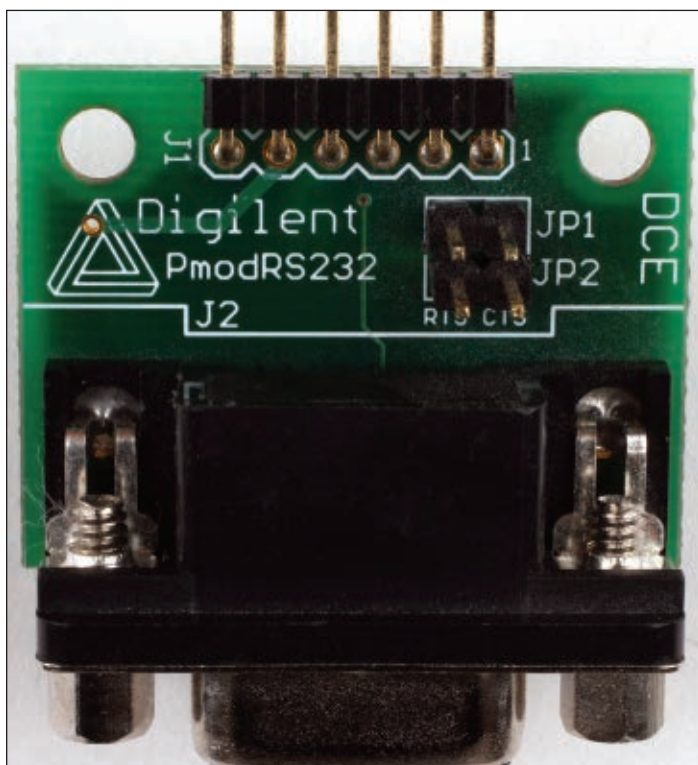


PHOTO 3. The PmodRS232 is one of a number of Digilent Pmods. If you had time to check out the recent *SERVO* Cerebot discussions, you're already familiar with Pmods. In this case, I've called up a PmodRS232 to save me from wiring an RS-232 interface from scratch.

verify the PIC32's UART interface and the code that will ultimately support it.

The Digilent PmodRS232 shown in **Photo 3** is an instant and portable RS-232 interface. Based on an ST3232 RS-232 converter, the PmodRS232 eliminates the need to build a true RS-232 portal from scratch. With VCC, GND, TX, and RX pinned out on a set of six pins, all I have to do is add the appropriate socket to the PIC32MX795F512L TRAINER, wire it into the PIC's

UART, and plug in the PmodRS232. Just like that, I've stirred up an instant and removable RS-232 interface. Once again, a diagram like the one you see drawn up in **Figure 2** is all we need to successfully utilize the services of the PmodRS232.

The PIC32 I/O pin RG6 can be configured to act as a RTS (Request To Send) signal for UART2A, while the PIC32's RG9 I/O pin doubles as UART2A's CTS (Clear To Send) signal. In our design, we will not activate UART2A's RTS and CTS functionality and use I/O pin RG6 to drive the μ OLED-128-G1's active-low open-collector RESET input.

UART2A Verification

The other old friend we'll use to test our UART2A interface is Tera Term Pro which I'll run on an XP Pro-controlled Lenovo NetBook. Tera Term Pro will interface to the PIC32MX795F512L's UART2A interface via the PmodRS232 module we've provided socket space for on our TRAINER. The Pmod-equipped RS-232 test rig is captured in **Photo 4**. As you can see in **Schematic 1**, the TEST ONLY PmodRS232 connector is tapping into the TX and RX lines that connect the UART2A interface and the μ OLED-128-G1 display module. The μ OLED is not mounted in the test configuration. Power for the PmodRS232 is provided by the TRAINER's on-board 3.3 volt LDO voltage regulator.

Bulk power for the TRAINER/PmodRS232 combo and eventually the μ OLED-128-G1 display module is sourced from a host PC's USB interface. In our design, the TRAINER's PIC32MX795F512L is running HID-class device code based on the MCHPFSUSB Framework. Although we could run our RS-232 test without the services of a HID-class device, there are a couple of good reasons for doing so.

We can use the USB enumeration process to control the TPS2041BDBVT switch which directly controls the presence or lack of presence of +5.0 volts on the TRAINER's four-layer printed circuit board's power plane. With control of the +5.0 volt power, we are able to control the power applied to the μ OLED-128-G1. Obtaining power by way of the USB portal also allows us to involve the USB descriptors. The HID-class device descriptors control the amount of current drawn from the host USB portal. Thus, we can programmatically dial in a current consumption cap for the TRAINER and the associated external peripherals it powers.

While having nothing to do with our test process, the availability of HID-class powers provides a high speed communications conduit between the TRAINER and its host device. The HID-class USB link gives the programmer access to the μ OLED-128-G1 from the host PC through the PIC32.

Now that USB has been justified in our design, the first order of business involves configuring the maximum current that is to be drawn from the host

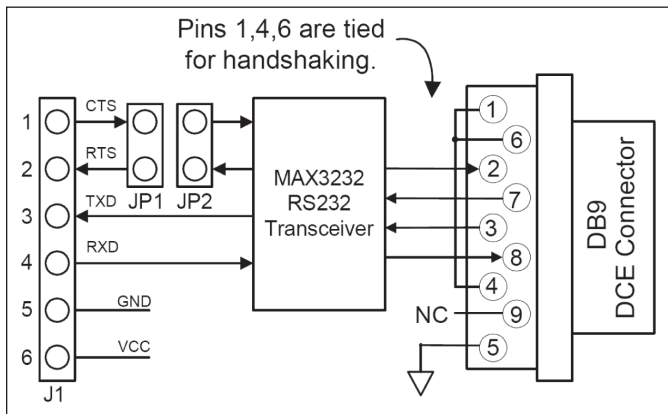


FIGURE 2. If you were here last month, you already know what a Pmod is. Well, here's another tricky little module from Digilent. The PmodRS232 is an instant RS-232 port that we can plug into our PIC32MX795F512L TRAINER as required.

USB portal. As I mentioned earlier, this is done in the HID-class device descriptor:

```
/* Configuration 1 Descriptor */
ROM BYTE configDescriptor1[]={
    /* Configuration Descriptor */
    0x09, //sizeof(USB_CFG_DSC),
    // Size of this descriptor in bytes
    USB_DESCRIPTOR_CONFIGURATION,
    // CONFIGURATION descriptor type
    0x29, 0x00,
    // Total length of data for this cfg
    1,
    // Number of interfaces in this cfg
    1,
    // Index value of this configuration
    0,
    // Configuration string index
    _DEFAULT | _SELF,
    // Attributes, see usb_device.h
    125,
    // Max power consumption (2X mA)
};
```

The descriptor code snippet is part of the PIC32MX795F512L's usb_descriptors.c file in the download package at www.servomagazine.com. As you can see, our current cap is 250 mA.

Next, we must specify UART2A as our UART module

interface of choice. The PIC32MX795F512L is supported by a comprehensive set of peripheral functions and macros which are assembled in the Microchip PIC32MX Peripheral Library. The Peripheral Library's function calls are defined in detail within the Library Help Files (which are part of the MPLAB C Compiler for PIC32 MCUs). All we have to do is include plib.h in our source code to gain access to the Library:

```
#include <plib.h>
//the following #defines are contained
within HardwareProfile.h
#define CLOCK_FREQ 80000000
//peripheral clock frequency
#define DESIRED_BAUDRATE (57600)
//the desired BaudRate

static void InitializeSystem(void)
//found in Nuts and Volts-
//pic32mx-hid.c
{
    //other initialization code
    //goes here

    UARTSetDataRate(UART2A, CLOCK_FREQ, DESIRED_
    BAUDRATE);
    UARTSetLineControl(UART2A, UART_DATA_SIZE_8_BI
    TS | UART_PARITY_NONE | UART_STOP_BITS_1);
    UARTEnable(UART2A, UART_ENABLE_FLAGS(UART_
    ENABLE | UART2A | UART_RX | UART_TX));
    mPORTGOpenDrainOpen(BIT_6);
    mPORTGSetBits(BIT_6);
}
```

The InitializeSystem function is part of the MCHPFSUSB Framework and is called first from the main function. I've called out the UART2 statements that were added to support UART2A. The UARTSetDataRate function uses the peripheral clock frequency – which happens to be our design clock frequency – to calculate the value to place into the UART2A U2ABRG register.

In **Screenshot 1**, I've captured all of the values that were loaded into the UART2A as a result of our calling the peripheral library functions. Let's see how well the UARTSetDataRate function worked. Here's the equation for calculating the baud rate when the U2ABRG value is known:

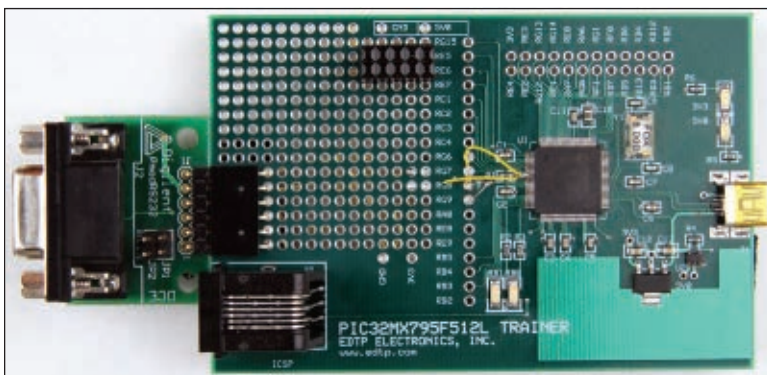
$$\text{Baud Rate} = \frac{F_{PB}}{16 \cdot (UxBRG + 1)}$$

Simple substitution will yield a baud rate value for us. If we assign FPB the CLOCK_FREQ value of 80000000 and punch in the U2ABRG value from **Screenshot 1** (86 decimal), we come up with a calculated baud rate value of 57471. Ideally, our value should be as close to our desired baud rate of 57600 as possible. To calculate the error percentage we use the following formula:

$$\% \text{Baud Rate Error} = ((\text{Calculated Baud Rate} - \text{Desired Baud Rate}) / \text{Desired Baud Rate}) * 100$$

Punching in the numbers on my trusty

PHOTO 4. Adding a socket for the PmodRS232 allows us to remove and reconfigure the RS-232 interface following successful testing. Depending on your application needs, being able to instantly add a serial port could be a plus, as well. As you can see, I've also wired in a 10-pin male header for the display module.



HP16C results in an error percentage of -0.22%. That should be good enough to reliably get bits to and from the Tera Term Pro application running on the Lenovo NetBook.

Getting the baud rate value right is only part of what we need to do to bring up a working RS-232 link. The µOLED-128-G1 wants to see RS-232 packets that contain eight bits of data, a single Stop bit, and no Parity bit. That's basically what the UARTSetLineControl function arguments are telling us will be configured. Take another look at **Screenshot 1**. However, this time give your attention to the U2AMODE SFR (Special Function Register). We're interested in bits 0, 1, and 2. Bits 1 and 2 — the PDSEL (Parity and Data Selection) bits — are clear which represents eight-bit data packets with no parity. Bit 0 is the Stop Selection bit and it is also clear, indicating 1 Stop bit is appended to each eight-bit data packet.

At this point, we should be good to go with UART2A. All that's left to do is enable the UART module which we do by calling the UARTEnable function. Bit 15 of the U2AMODE SFR is the UART2A Enable bit which shows as set in **Screenshot 1**. When set, the UART2A Enable bit turns UART2A "ON" and places the UART2A pins under the control of the UEN (UART2A Enable) bits. The UEN bits are located at bit positions 8 and 9 in the U2AMODE SFR. The default condition of the UEN bits is cleared which enables the U2A TX and RX pins, and disables the UART2A RTS and CTS pin functionality. This configuration is desired as we are planning to use the I/O pin that doubles as UART2's RTS signal for the µOLED-128-G1's RESET pin driver.

Look away from UART2A's SFRs for a moment and focus on the U2BMODE SFR value. The only bit that is set in the U2BMODE SFR is the FRZ bit which is invoked during Debug mode to indicate that the operation is frozen when the CPU is in Debug Exception mode. In that I was stopped in Debug mode when I snapped **Screenshot 1**, the state of the FRZ bit is logical. I point this out as the FRZ bit in U2AMODE is also set and has nothing to do with the normal operation of UART2A. Note also that UART2B is Off, as bit 15 of U2BMODE is clear.

The UART_RX and UART_TX arguments of the UARTEnable function affect bits 10 and 12 of the U2ASTA SFR. Bit 10 reflects the ON state of the UART2A transmitter, while bit 12 displays the ON status of the UART2A receiver. Both bits 10 and 12 are set indicating that the UART2A transmitter and receiver are online.

The PmodRS232 connects to the PIC32MX795F512L's UART2A interface on its TTL side and channels through to the Tera Term Pro application on the RS-232 side. UART2A is configured and ready to go. So, let's insert this code snippet into the ProcessIO function of the MCHPFSUSB Framework HID-class device application code:

```
char rxdata;
if (UARTReceivedDataIsAvailable(UART2A))
{
    rxdata = UARTGetDataByte(UART2A);
    if (UARTTransmitterIsReady(UART2A))
    {
```

Virtual	Name	Hex
BF80_6400	U2AMODE	0x0000C000
BF80_6410	U2ASTA	0x00001510
BF80_6420	U2ATXREG	0x00000000
BF80_6430	U2ARXREG	0x00000000
BF80_6440	U2ABRG	0x00000056
BF80_6600	U2BMODE	0x00004000
BF80_6610	U2BSTA	0x00000110
BF80_6620	U2BTXREG	0x00000000
BF80_6630	U2BRXREG	0x00000000
BF80_6640	U2BBRG	0x00000000

SCREENSHOT 1. A PIC32MX Family Reference Manual and the bits in the PIC32MX795F512L's SFRs are all we need to figure out what's going on with the PIC32's six UARTs.

```
        UARTSendDataByte(UART2A, rxdata);
    }
}
```

Do you see what the code snippet is doing? Let's talk through it. If valid data has been received by UART2A, it is transferred from the UART2A receive buffer into the eight-bit rxdata variable. If the UART2A transmitter is idle, the byte in the rxdata variable is loaded into the UART2A transmit buffer. Upon seeing the incoming byte, UART2A serializes and transmits the byte. You got it! It's a character echo function. The good news is that I was able to echo characters from the Tera Term Pro session, through the PIC32MX795F512L, and back to the Tera Term Pro window.

Planting Pixels

With the PIC32 UART2 module operation verified, I rewired the PmodRS232-to-PIC32MX795F512L interface and added a two-position DIP switch (SW1) to isolate the UART from the PmodRS232 and µOLED-128-G1 serial interfaces. Note that the PmodRS232 and µOLED interfaces are always electrically connected. To use the Pmod/µOLED-128-G1 connection, both poles of SW1 must be in the OFF position which isolates the UART2 interface. With the PIC UART2 out of the circuit, we are able to communicate with the µOLED via RS-232. This arrangement allows us to load a PmmC or control the µOLED from a PC serial port. When both poles of SW1 are in the ON position, the µOLED-128-G1 is under the influence of the PIC32MX795F512L. The PmodRS232 module should not be mounted in its socket when SW1 is allowing an electrical connection between the UART2 and the µOLED.

Driving the µOLED with the PIC32 is a piece of cake. The µOLED-128-G1 requires at least one second of quiet time when you power it up. The power-up delay allows the µOLED-128-G1 to initialize its internal graphic memory and display registers. We are not allowed to communicate with the µOLED during this one second power-up delay time. Once the µOLED-128-



PHOTO 5. Note the absence of the PmodRS232 and presence of text on the display. This tells us that the PIC32's UART2 is driving the display module.

G1 has been given enough time to stabilize, we must instruct the PIC32MX795F512L to transmit a 0x55. The μ OLED uses the 0x55 to determine the serial link baud rate. Once it has zeroed in on the baud rate, it will return an ACK (0x06) to the PIC32MX795F512L. At this point, the μ OLED-128-G1 is ready for work.

We're going to need a timer to count down the one second power-up delay. A timer may also come in handy later if we find that we need to time other events in the application. Rather than write a specific timer routine for every unique event, we'll implement our timer using an interrupt. For now, a timer granularity of 1 mS will suffice. Here's the code we'll need to implement a 1 mS-per-tick timer:

```
OpenTimer1(T1_ON | T1_SOURCE_INT | T1_PS_1_8,
10000);
ConfigIntTimer1(T1_INT_ON | T1_INT_PRIOR_2);
void __ISR(_TIMER_1_VECTOR, ipl2)
Timer1Handler(void)
{
    // clear the interrupt flag
    mT1ClearIntFlag();

    if(++msecs1 == 1000)
    {
        ++secs1;
        msecs1 = 0;
    }
}
```

Basically, we're prescaling the incoming 80 MHz clock signal to Timer1 by eight, which results in a Timer1 tick of 10 MHz or 100 nS. On every 10,000 ticks (1 mS), we trigger an interrupt. The interrupt handler clears the Timer1 interrupt flag and accumulates the 1 mS ticks into seconds. Now that we have an accurate way to measure time in seconds, we can write the μ OLED-128-G1 power-up delay code:

```
msecs1 = 0;
secs1 = 0;
LATFCLR = p5V_ENABLE;
while(secs1 < 2);
```

After clearing the 1 mS timer variables, we turn on

the TPS2041 which allows +5.0 volts to flow to the μ OLED-128-G1, and then wait for one second to pass. At this point, the μ OLED-128-G1 should be ready to talk to us. So, we transmit a 0x55 to it and wait for an acknowledgment:

```
rxdata = 0;
do{
    UARTSendDataByte(UART2A, 0x55);
    msecs1 = 0;
    while(msecs1 < 11);
    if (UARTReceivedData
    IsAvailable(UART2A))
        rxdata = UARTGetDataByte(UART2A);
}while(rxdata != dmACK);
```

Once the 0x55 has been sent, we must give the μ OLED-128-G1 a bit of time to sort things out before looking for a response. As you can see, we have put the 1 mS timer to work on a 10 mS delay before checking for an ACK from the μ OLED-128-G1.

If the μ OLED responds with an ACK, it has synced up with UART2 and is ready to receive and process commands. What do you think this code will do for us?

```
unsigned char cntnr;
unsigned char cPutSERVO[] =
{0x73,0x00,0x00,0x02,0xFF,0xFF,'S','E','R','V',
',','O',0x20,'M','A','G','A','Z','I','N','E',','>
',0x00};

for(cntnr=0;cntnr<sizeof(cPutSERVO);++cntnr)
{
    while(!UARTTransmitterIsReady
    UART2A));
    UARTSendDataByte(UART2A,
    cPutSERVO[cntnr]);
}
while(!UARTTransmissionHasCompleted
(UART2A));
```

The results of executing the Place String of ASCII Text (Formatted) command (0x73) can be seen in **Photo 5**. The text to be displayed begins at column 0x00, row 0x00, and will be shown in an 8x12 font (0x02). The displayed text color is white (0xFF,0xFF). The ASCII string must be terminated with a 0x00. Pretty cool, huh?

Stay Tuned

As the Carpenters would say, "We've Only Just Begun." I've got more μ OLED stuff on tap. So, we'll scatter some more PIC32MX795F512L and μ OLED-128-G1 floobydust next month. **SV**

Fred Eady may be reached via email at fred@edtp.com.

Sources

μ OLED-128-G1 Display Module
Saelig
www.saelig.com

PmodRS232
Digilent
www.digilentinc.com

PIC32MX795F512L
MCHPFSUSB Framework
Microchip
www.microchip.com

PIC32MX795F512L
TRAINER
EDTP Electronics, Inc.
www.edtp.com

The Bioloid Premium Kit



It's time to start your own robot evolution.

Recently, I had the opportunity to review ROBOTIS' latest offering: the Bioloid Premium kit. I have spent many years building my own robots. A few years back, I discovered Robo-Cup and noticed that nearly all the leading teams in the humanoid league were using ROBOTIS servos or Dynamixels. After studying the Dynamixel's capabilities, it became apparent to me why they seemed so popular in the collegiate circles. Dynamixels are intelligent servos that can be daisy-chained on a high speed serial bus. Each unit is individually addressable with its own ID. They can be configured separately, sent individual commands, or several can be synchronized with a single command. ROBOTIS provides detailed specifications of registers and the communication structure of the Dynamixels thus enabling others to easily use these building blocks to create and program robots of their own. The Bioloid platform provides an excellent opportunity for learning and entertainment to those that are just starting out and also to those who are already seasoned roboticists. In this article, I'll cover details of the hardware and assembly in the new kit, and in a follow-up article I'll cover the new software ROBOTIS has included.

By Rob Farrell

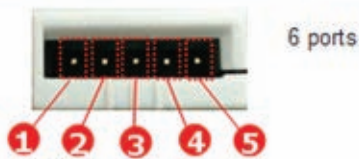
Mind

The new CM-510 controller is the brains of the robot and it has some very nice improvements over the original CM-5. The CM-510 is based on the ATmega 2561 which is virtually identical to the CM-5's ATmega128 except it has 256K bytes of Flash instead of 128K. This means more room for firmware, motions, and tasks. In fact, ROBOTIS has partitioned the Flash memory to have twice the motion pages as the CM-5. The CM-510 also has six general-purpose analog input ports. The 5V output will allow sensors that have high current sources such as an LED illuminator in an IR reflectance sensor to be switched off when not in use to conserve battery power. It can also be used to light external LEDs. The analog inputs are keyed and clearly marked as to their port designation.

On the back of the controller are mode and wireless



The CM-510 A.



6 ports

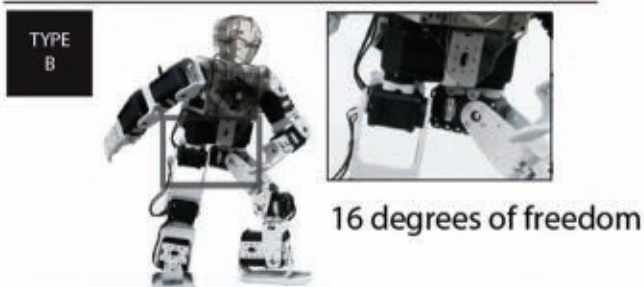
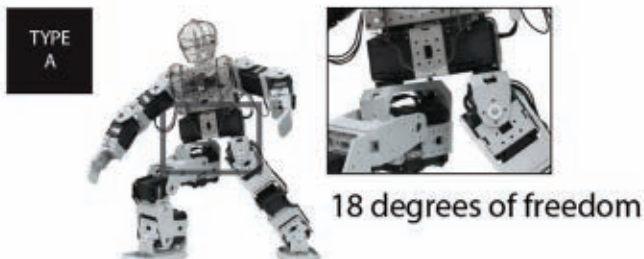
1. OUT : 5V Output
2. VCC (5V)
3. ADC : (10bit) Input
4. GND
5. NC : Not used

CM-510
General-Purpose
Analog Parts.

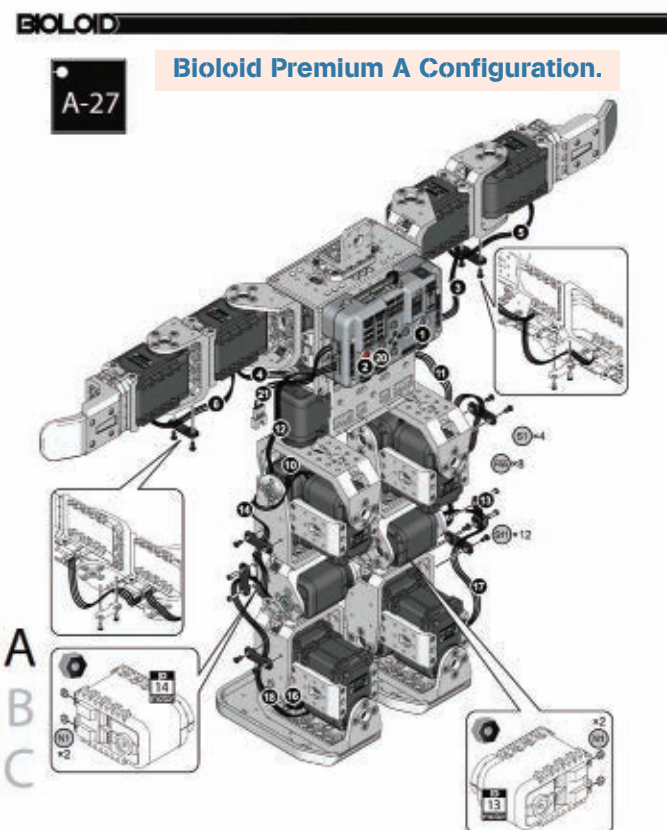
ex: simple LED circuit using the output line



Bioid
Premium
Assembly.



Three Humanoid Models.



communication indicator LEDs, along with the familiar direction, start, and mode buttons found on the CM-5 controller, as well. The mode button is used to select from manage, program, and play modes. The remaining five buttons are available for inputs to tasks or behaviors. In the default behavior program, the buttons are used to select from the many available tasks or behaviors which ROBOTIS offers. On the top of the controller is a stereo mini audio jack that provides a direct serial connection with a nine-pin RS-232 port or via a USB port with the included USB2Dynamixel adapter. The CM-510 is much

slimmer than its predecessor, largely because the battery pack has been removed and separated from the controller and is now

contained in its own easily removable housing. Included in the CM-510 is a small microphone that serves as a sound

detector for things like counting claps within a period of time. It also has a register that can count. This can be useful to select a task or motion. Adding your own external sensors such as tilt, FSRs, accelerometers, and other analog output sensors will be easy.

Body

Where the original Bioloid kit is configured to be a set of modules that can be built into many different robot configurations, the new Premium kit is clearly focused on humanoid models. The AX-12+ Dynamixels and most of the brackets remain the same as the original Bioloid, so clearly many configurations can still be built from this kit. Most of the physical changes from the original kit are in the chest and head. In the Quick Start Guide, there are assembly instructions for three humanoid configurations. I chose to build the type 'A' which seemed to be the most versatile one.

Overall, it took me three afternoons to build up my type A humanoid. As I progressed through the Guide, I was impressed at the level of detail in the Premium Kit's instructions. They contain few words but provide highly detailed graphics. I've built a number of other kits and it

seems that I always find some inconsistency in the instructions which causes me to second guess things as I go through it. A couple of times I thought I found a mistake, but in the end the only mistakes I found were my own. From the feet up, this was an excellent build. No cheesy parts that break easily or impossible assemblies. I think the most difficult part of the assembly was inserting the tiny 2 mm nuts into the servo nut holders and brackets. (After a little practice, it goes quickly.)

Sensors

The Premium Kit comes with many sensors. The Dynamixels themselves have the ability to measure and report their current positions, speed, load, voltage, and temperature. As mentioned earlier, the CM-510 contains a sound sensor that can be used to initiate tasks. The Premium

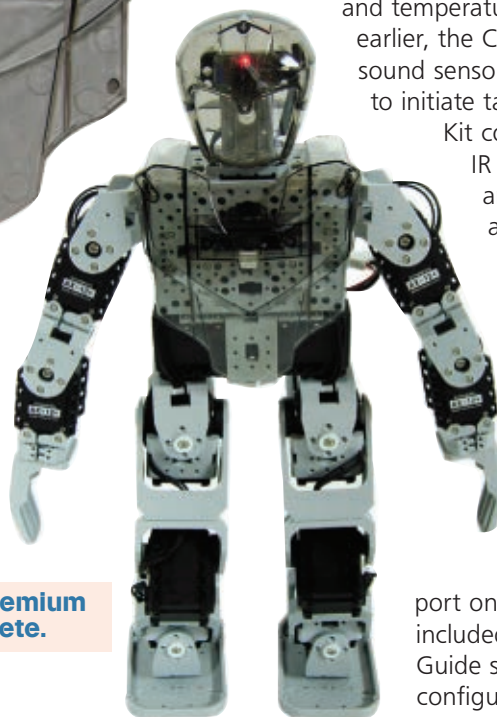
Kit comes with three IR sensors: two that are short range and one that can be mounted in the chest that is precise enough to measure distance. Each of these IR sensors plugs into an available general-purpose analog port on the CM-510. The included Quick Start Guide shows a configuration where the short range IR sensors are used on the feet. I

imagine they could be used for automatic ball detection for soccer. I chose not to install the short range sensors at least initially but may later as I am a big fan of robot soccer. Also included is a two axis gyro. The gyro requires two analog ports and is easily mounted in the chest cavity on small barrel support posts.

Control of the robot is achieved with the included gamepad or RC-100 which is often referred to as a boomerang controller. The RC-100 has 10 buttons that can be used as inputs in any combination which provides for many possibilities. The default configuration for the RC-100 is IR, however, I've been told that the first 1,000 Bioloid Premium kits will ship with the Zigbee wireless upgrade.



Body Shells.



Bioloid Premium Complete.

IR Sensors.



The Zigbee upgrade includes a transmitter that gets installed into the RC-100, and a transceiver that gets mounted in the humanoid's head and then is attached to the wireless port on top of the CM-510 controller. Installing the Zigbee transmitter in the boomerang controller presumably works by insertion of the module pins into the open-plated through holes in the RC-100 circuit board. Personally, I wasn't satisfied with a friction fit here and soldered my module in so there was no chance of it coming loose and interrupting commands.

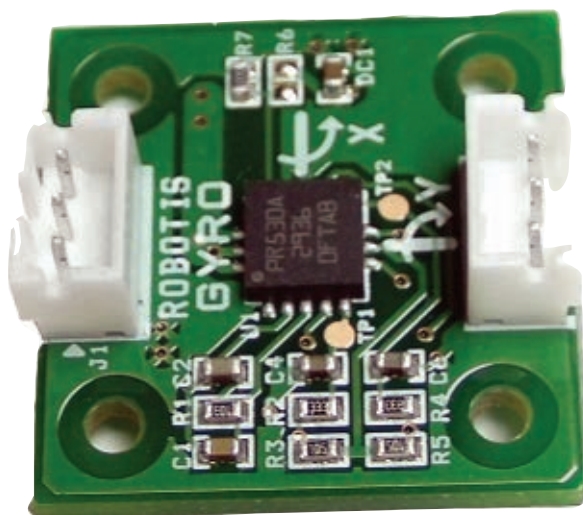
Control can also be achieved using the included serial cable attached to a PC, either directly or with a nine-pin serial port via the included USB2Dynamixel. I've been told that soldering the Zigbee module in is not necessary but it made sense for me.

The USB2Dynamixel included in the kit is a handy little device that serves a couple



Boomerang Controller.

purposes. Most simply, it's a USB-to-serial converter that can be used to interface a PC to the CM-510 controller for programming or control. It also has a small switch on the side that allows it to be configured to communicate to the Dynamixels directly without the CM-510 controller. This is another way to operate your robot as a slave from a PC.



Two-axis gyro.

Every robot should have a kill switch so that if the robot behaves unpredictably it can be shut off immediately. On the back of the CM-510 controller is a small black power switch.

Lithium polymer batteries are very high energy densities and many products don't use them because of their inherent danger from being damaged, over-discharged, or improperly charged. They can easily create fires. The battery included in the Premium Kit is a three cell 11.1V 1,000 mAH lithium polymer which has a self-contained power protection circuit that limits operating voltage, and maximum charge and discharge current. The battery is mounted in a plastic shell that should protect it from getting damaged in any robot battles or falls. To charge the

battery, it's disconnected from the robot via its four-pin plug and connected to the included Yuntong balance charger and DC power supply. The balance charger charges each



ZigBee Modules.



More commonly, the Dynamixel interface is used for testing and configuration of single Dynamixels using the Dynamixel Manager application.

Safety needs to be the top priority with any robot – 18 high powered geared motors and wires in motion powered by high energy density lithium polymer batteries ... what could go wrong? Fortunately, ROBOTIS has built many safeguards into the new Bioloid Premium Kit. First and foremost is the kill switch.

cell separately so that each cell receives the appropriate charge, thus keeping them in balance.

To protect the system from excessive current, a small 5A fuse is included on the back of the CM-510 controller. Should a wire short out or a motor fail, or if the robot is

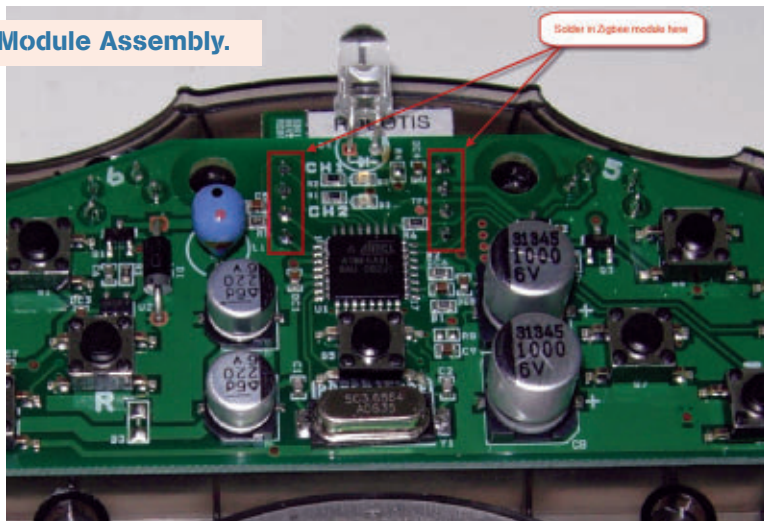
given instructions that strain many servos at once, the fuse will blow. The fuse is easily accessible once the CM-510 is removed from the robot. ROBOTIS includes an extra fuse. (The fuses are common and easily found on popular electronics websites.)

Each Dynamixel has safeguards of its own, too. Limit temperatures and voltages are part of its configuration. If limits are exceeded, then it automatically shuts down and flashes its indicator LED.

Proper wiring and regular inspection is another important safeguard to keep your robot safe and functional. The Quick Start Guide provides excellent detail on wiring the three humanoid configurations and in the kit are some cable holders that easily screw to the sides of brackets to help keep the robot's wires out of harm's way. When I first assembled my humanoid, I tried to keep the cables less visible but later decided that I would route them for ease of repair instead. I ended up with a bit of a compromise between what is suggested in the Guide and my attempts to reduce the cable visibility.

Periodic maintenance is required to keep your robot operating properly. With 18 degrees of freedom and lots of vibration from motion, some screws will come loose. I find it's a good idea to check screws on critical joints after every run. The screws that frequently get loose in my robot are the center screws in the freewheeling bushings. Those in the legs get loose more often, but I find it's a good practice to check them all. In the past, I have used Loctite on screws like these but after much experience, I would recommend against it as it can make things very difficult to disassemble. Perhaps a lock washer could be more effective at keeping things tight.

Zigbee Module Assembly.



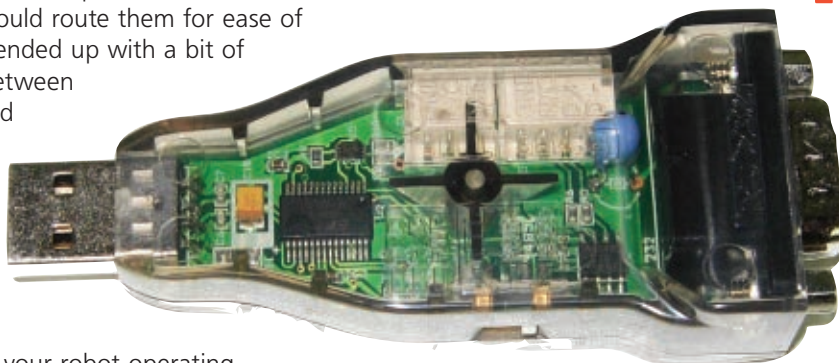
Final Thoughts

It's clear that ROBOTIS has listened to their customers. They've focused on the humanoid model with the changes and additions to the chest, but overall it's not exclusively humanoid. ROBOTIS has added external analog inputs and sensors, and even support to attach your own sensors.

They've added a high performance battery and charger. The system remains highly modular and reliable, safe, and expandable. There are significant improvements to the Bioloid hardware but the really big improvements have been in the software. I'll cover more details on RoboPlus (the new Bioloid software) in an upcoming article.

I heard about this new kit more than a year ago and I can say it's been worth the wait. Assembling the Bioloid and even mastering the new RoboPlus software is really just the starting point. I have come to think of ROBOTIS not as a company who builds robots or robot kits but rather a company that enables others to become roboticists. What they provide is the stepping out point. Join the evolution. **SV**

The DSCN 3119.

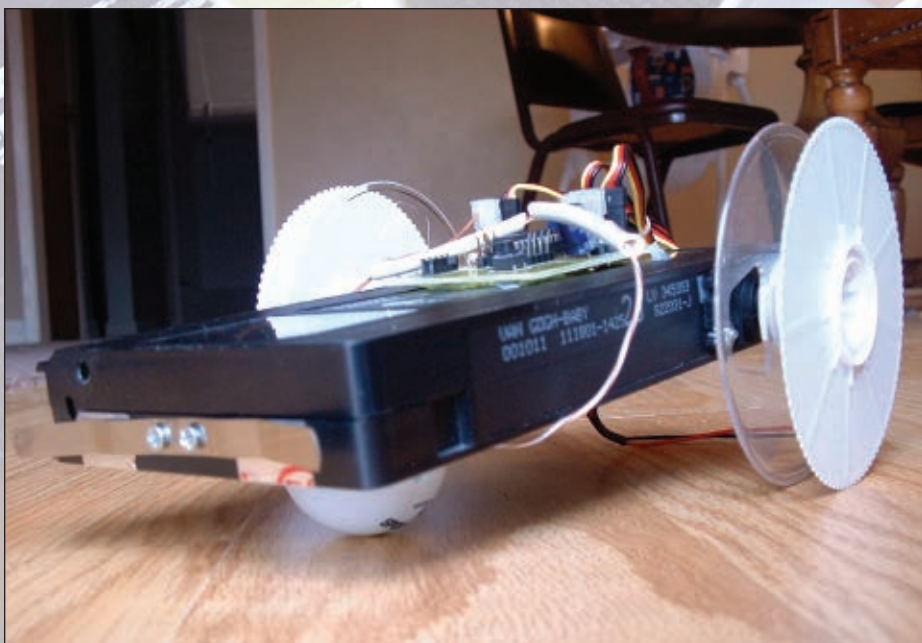


Charger and Battery.

THE CASSETTE BOT

By Bryan Jackson

How do you make a robot out of a VHS cassette tape? What is a VHS cassette tape? Is this guy cheap or what? These questions and many more will be answered in this article about making a cheap VHS cassette robot.



It all started with a bang. My wife and kids had been to Grandma's house, and after a good day of terrorizing the grandparents, had brought home some borrowed video tapes. (Those are the things your parents used before DVDs were invented.) Immediately upon entering the house, my darling children accidentally dropped several of the video cassettes on the floor resulting in one VHS cassette suffering damage from a broken dust flap. Immediately after the obligatory lecture about being careful with other people's stuff, several pairs of expectant eyes turned toward me.

"Can you fix it?" my wife asked.

"If you have a cassette you don't care about I can pull parts from it," I replied.

So, I took the sacrificial tape apart and salvaged the aforementioned dust flap. I then expertly replaced the broken part on the offending tape and put everything back together again. Of course, it didn't play so I took it apart again and put everything in the right place this time, and got the thing working (thus remaining in the in-law's will and on their good side for at least the next week).

So, what do I do with the rest of the parts? Why make a robot, of course. It was so obvious looking at the parts strewn across the table. The small imp in charge of VHS

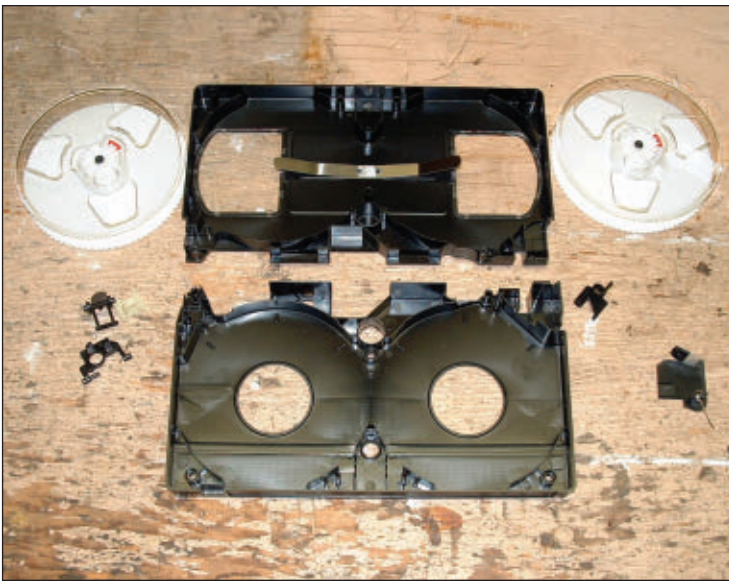


Figure 1. The disassembled VHS cassette.

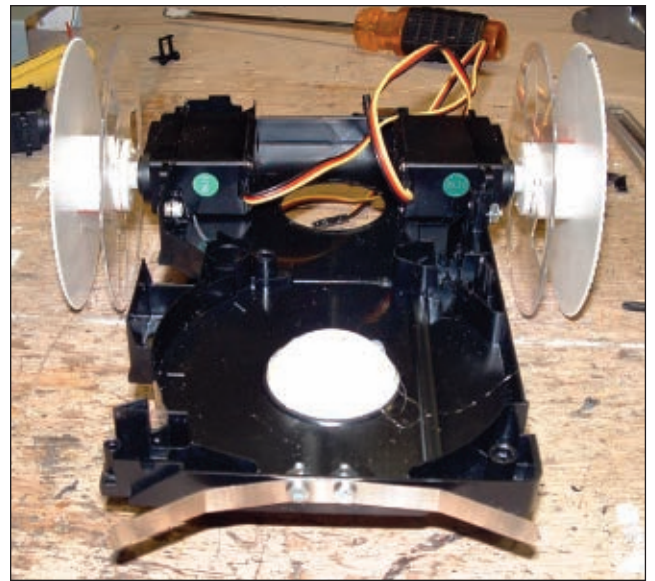


Figure 2. Open cassette case showing mounted motors and sensor.

tapes probably planned it that way. The two tape reels were the perfect size for wheels. The body of the tape is a good size for the frame and there is even a long strip of springy steel that would make a perfect bump sensor (**Figure 1**).

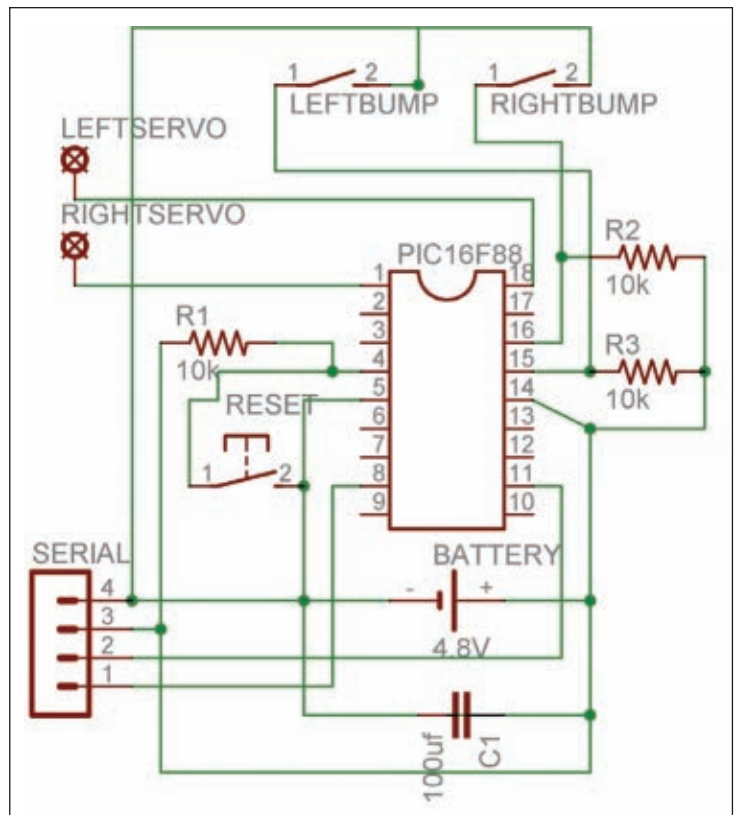
The first step was to rip out all the tape from the reels. This takes a surprising amount of time and you may want to enlist the finely honed skills of a two-year-old child in this endeavor. Once the reels are empty, cut the tape from the reels and throw it away. The rest is a matter of personal style and available materials.

I first pulled some servos from an old robot frame. These servos had already been modified for continuous rotation. I cut rectangular holes in the side of the cassette tape body for the servos and drilled a couple of holes for the mounting bolts. Next, I drilled a small hole in the center of each reel. I bolted the servos to the body, and then removed the screw from the servo horn and used it to screw the reel to the servo. (Rinse and repeat for the other side.) A ping pong ball hot glued to the hole in the bottom front of the body is the third "wheel." This works quite well for a small light robot that generally stays on smooth surfaces.

The bump sensor is very simple but effective. I drilled a couple of holes in the front of the case using the holes in the spring strip as a guide. Then, I bolted the strip to the front of the case (**Figure 2**). I used a wire under the nut to connect the strip to ground on the controller. For the other half of the bump switch, I used duct tape (the aluminum tape for real duct work). If you don't have a roll, you need one now. Just remember the glue on the sticky side is a pretty good insulator, so make connections on the top. I cut a couple of strips, stuck them under the bump sensor spring on each side, and ran the excess under the body of the bot. Then, I used another piece of tape to stick the

stripped end of a wire to the duct tape (**Figure 3**). A pull-up resistor and some wiring complete the sensor (**Schematic 1**).

The battery is from a 9.6V rechargeable pack that I cut in half lengthwise to make a flat 2x2 4.8V pack. I have had good success running 5V microcontrollers from a 4.8V NiCd pack without any voltage regulation.



Schematic 1.

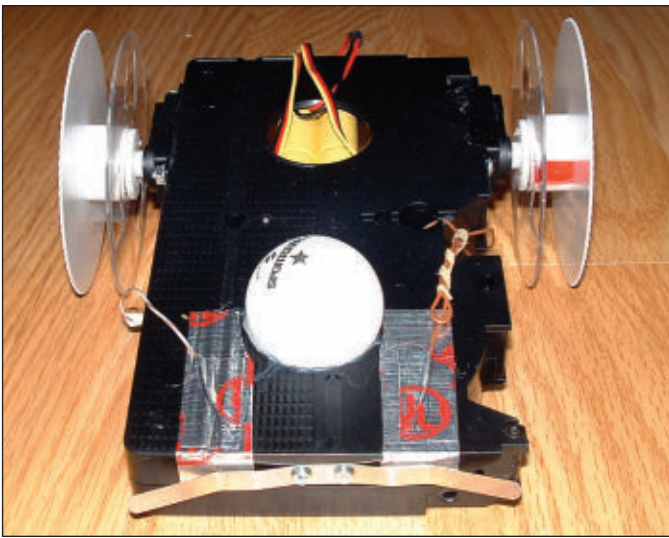


Figure 3. Bottom view showing bump sensor connections.

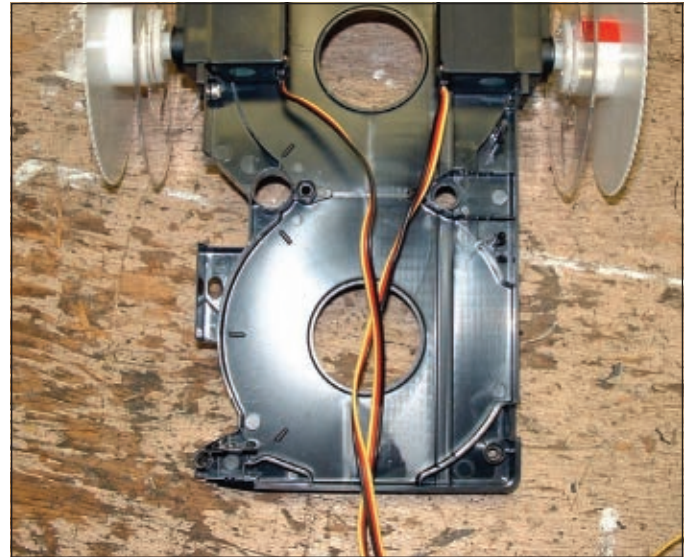
The controller is a PIC16F88 loaded with the Tiny bootloader and programmed using a MAX232a TTL to RS-232 converter. The circuit board was left over from another project, and fit quite nicely. Schematics and artwork for the board can be found on my website (www.highschoolrobots.com).

Programming was quick and dirty. I use JAL (Just Another Language), and Tiny bootloader as my development platform. I like the JAL compiler because it's free, and the code is easy to read. The self-programming capability of the 16F88 coupled with a bootloader make it very easy to program. The source code for this project is also available on my website.

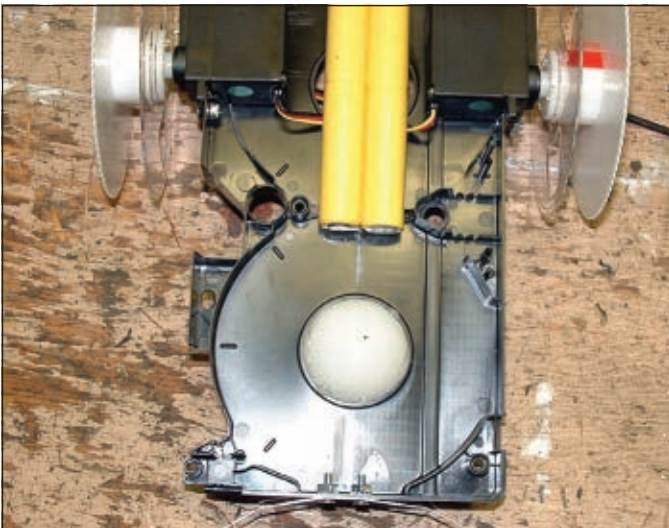
After half an hour or so, I had the code working and the bot was happily bumping into stuff. The code is pretty simple. I just have the bot go forward until one of the sensors is tripped, then back up a little and turn in the opposite direction of the sensor that was bumped. I noticed



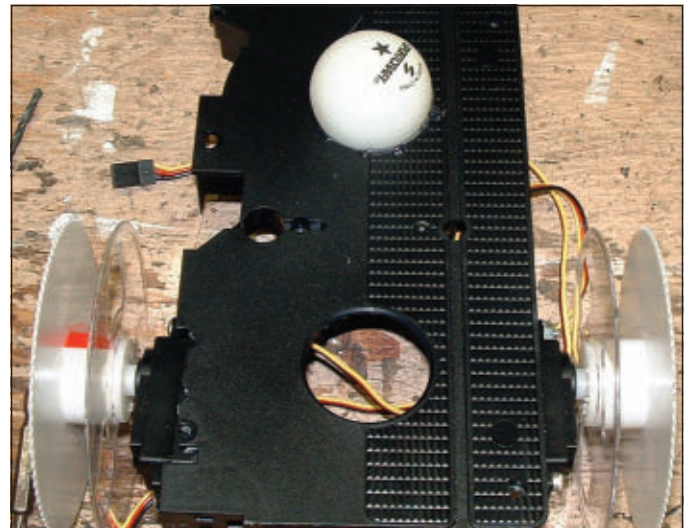
Side view showing servo motor mount.



Open cassette case showing motors.



Open cassette case with battery pack.

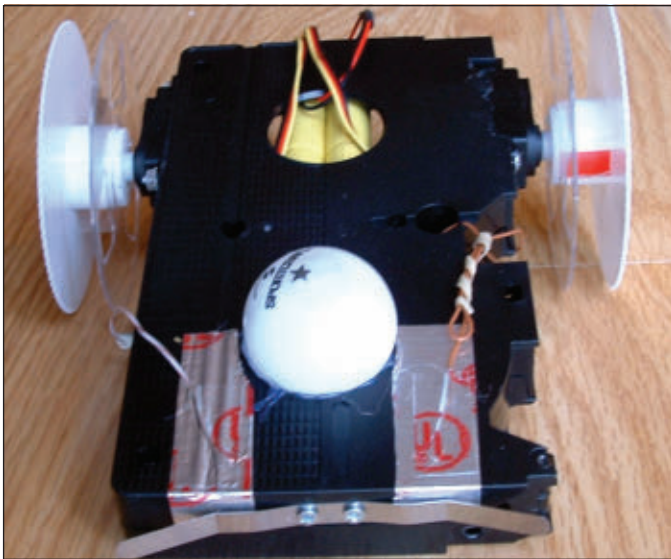


Bottom view showing ping pong ball.

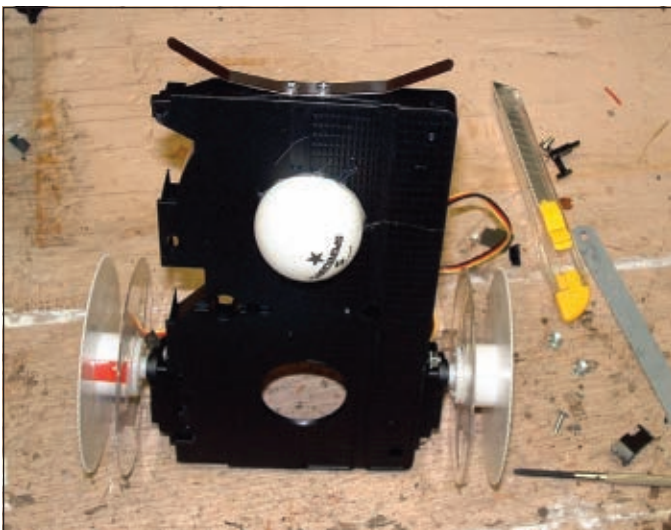
that the bump sensor was a little too stiff, so I pulled it off and put a couple extra bends in it to put the contact portion closer to the duct tape. This helped quite a bit. The bot still gets stuck if it hits a corner dead center, and it can get stuck with a wheel on a chair leg and drive circles around the leg forever.

My next modification will be to extend the bump sensor out to cover the wheels and maybe put something in the middle to get rid of that blind spot.

It was a fun little project and the kids got a kick out of it. They had a lot of fun poking the bump sensor to make the car back up and turn away. This little bot is capable of quite a lot of modification. There is plenty more space inside the case. The clear inside disk of the wheels is screaming for some sort of optical odometry; the clear windows on top need some sort of display; and, of course, it needs more sensors. I wonder what I did with that old Polaroid camera with the sonar range finder ... **SV**



Bottom view of completed car.



Bottom view showing bump sensor.

From the Smiley's Workshop Arduino Projects Kit

- Blink LEDs (Cylon Eyes)
- Read pushbutton and 8-bit switch
- Sense Voltage, Light, and Temperature
- Make Music on a piezo element
- Sense edges and gray levels
- Optically isolate voltages
- Fade LED with PWM
- Control Motor Speed

Only \$79.95

Visit our webstore @ www.nutsvolts.com for complete details.



AP CIRCUITS
PCB Fabrication Since 1984

As low as...

\$9.95
each!

Two Boards
Two Layers
Two Masks
One Legend

Unmasked boards ship next day!

www.apcircuits.com



The Typewriter Repairmen 2009 NURC ROV:

By David Forbes, Steven Forbes, Jim Forbes, and Janet Forbes



Inspired by a previous visit to the NURC event and an overwhelming desire to build an effective submersible robot, we formed a family team to compete in the 2009 National Underwater Robotics Challenge. Our team, “The Typewriter Repairmen,” is comprised of David, Jim, Janet, and Steven Forbes. Our Robot — “notBob” — is the result of three months of designing, building, and testing.

NURC

The National Underwater Robotics Challenge (NURC) is an annual competition put on by the Arizona Promoters of Applied Science in Education (APASE). The competition is unique in that it is open to nearly everyone. Elementary schools, high schools, colleges, and independent or corporate teams all compete to complete the same mission tasks. Points are awarded based on the completion of individual mission tasks, as well as on a report and presentation.

Each year features a new mission scenario, ranging from retrieving torpedoes from a sunken U2 submarine to taking depth and temperature measurements at an underwater ice core research outpost. A typical mission has a time limit of 30 minutes, and more than enough tasks to keep a team occupied. As an added twist, all of the missions take place at night to simulate the lack of sunlight at extreme depths, since making a robot that simply survives underwater isn't challenging enough!

The 2009 NURC scenario required teams to take a temperature reading and sound recording of a volcanic vent, retrieve underwater ice core and bacteria samples in the form of glow sticks, free another trapped ROV, and deploy a new science package to an underwater research station.

ROV Design

We spent a considerable amount of time analyzing the requirements and figuring out precisely how the robot would need to function to complete all of the given tasks. Complexity is not something you want in a system that operates in a wet environment — where a small failure can mean the loss of a robot — so we made every effort to simplify the design as far as possible while keeping our desired functionality.

The final ROV is shown to the left. Our analysis of the problem led us to believe that being able to move and see well were the two most important factors to a winning

design. Four thrusters give the ROV full mobility, while a stereoscopic vision system with tilting capability gives the driver the ability to see well in nearly any direction. LED flood lights provide good illumination of close objects, while a 20 watt halogen spotlight can be used to see far away objects; the lights tilt with the cameras.

To eliminate excess complexity, we decided that the cameras, lights, and entire control system should fit into a single watertight cylindrical compartment (SCULL) at the front of the robot. Thrusters made from RULE 1,100 gph bilge pump motors and model boat propellers (as seen in **Figure 1**) make construction easy and require virtually no waterproofing effort. We also determined that a manipulator mechanism with a single degree of freedom (opening/closing) would suffice and could be driven by a bilge pump motor.

The ROV is controlled with two Vex transmitters which are connected through their rear tether ports to the Hello Kitty™ interface panel. The sealed lead acid batteries and solar charge controller also connect to the interface panel, along with the video and audio signals to the television monitors (**Figure 2**).

NotBob Control Electronics

The ROV requires an electronic control system to accept commands from a poolside control box and convert them into motor drive signals, light switches, and other such functions. We had to decide on a control system that would fit in the small space available in the SCULL and provide control for five motors, a servo, and a 20 watt light bulb.

We evaluated the available electronic speed controllers for 12V brushed DC motors. All the units examined — for example the BaneBots BB-12-45 — accept 12V and feed it to a motor with a PWM duty cycle directly related to stick position. This is fine for a system with the battery nearby, but we would need an eight gauge tether cable to provide enough power for all the motors from a 12 volt battery on the surface. Alternatively, we could use a 25 amp-hour 12 volt battery in the ROV, which would double the weight.



FIGURE 1.

One of four thruster assemblies on the ROV, consisting of a RULE 1,100 gph bilge pump motor, Dumas 3" brass prop, machined propeller shaft, and a Kort nozzle constructed from ABS pipe.

One of us (David) is an electrical engineer with a bit of experience in motor control, having spent the previous summer rebuilding a precision mirror positioning system for a radio telescope at his day job. This servo system uses 48 volts to run a pair of 12 volt servo motors used to rotate a mirror to a specified angle.

How is it possible to run a 12V motor from 48 volts? The answer is simple: a lower PWM duty cycle. The motor current increases linearly with duty cycle, but the current also increases linearly with applied voltage since the formula for an inductor is $di/dt = V/L$. If you supply four times the voltage but a quarter of the duty cycle, the motor runs at its nominal rated current.

By using a higher supply voltage and reducing the duty cycle proportionally, it's possible to run the motor at its rated current while only supplying that current for a fraction of the time. This provides a big advantage for remotely powered motors. The lower duty cycle means that the overall supply current may be shared *in time* between several motors, allowing several 10 amp motors to run from a single 10 amp supply. (See **Figure 3**.)

The effects of resistive cable loss are reduced by the square of the voltage increase, so a much smaller tether cable can be used at 48V. We used 16 gauge speaker cable for the power supply to run four 10 ampere motors located 75 feet from the battery box.

One drawback is that there is no longer a source of 12V onboard the ROV, since the tether provides 48V. This is solved with a 12V DC-DC converter consisting of a National Simple Switcher chip and a coil, two capacitors, and a diode.

A motor speed controller circuit board — the ESC4 (**Figure 4**) — was designed to provide four proportional PWM speed controllers capable of 10 amperes each, all

driven from one 48V 10 amp source. Two of these boards are used in the ROV. The second board drives only the manipulator motor, so its current draw is minimal. A web page describing this board in detail, with full schematic diagrams, source code, and PC board layout files is in the **sidebar**.

The motor drive circuit (see **Figure 5**) is called an "H bridge" because the shape of the circuit diagram resembles the letter H. Two MOSFET transistors supply current from the 48V rail to either end of the motor, depending on the desired direction. Two more transistors sink the current from the opposite end of the motor, or they may both be turned on while the top two are turned off to short out the motor during the OFF phase of the PWM cycle. Otherwise, the current would have to build up from zero on every PWM cycle which is impossible for high switching rates (above 1 kHz) due to the coil inductance.

The MOSFET transistors have a 12V gate drive requirement to turn on the transistor to its low $R_{ds(on)}$ resistance. The CPLD that controls the transistors puts out only 3.3V. Fortunately, there is a part called a half bridge driver IC that drives the gates of one set of high and low side MOSFETs. It requires a 12 VDC power source and accepts two logic-level gate enable signals. The half bridge driver contains an internal level shifter to make the needed 12V gate voltage (relative to the source pin) for the high side transistor.

We need to sense motor current to ensure that the motor doesn't burn out if stalled. A sense resistor on each low side switch source develops voltage proportional to the current. The 0.02 ohm resistor gives 0.2V for 10A current.

A pair of comparators in the LM393 monitor the voltage across the two sense resistors. Two comparators are used because either low-side switch may have the motor

FIGURE 2.

The land-based portion of the control system, consisting of battery box (with solar charging controller), Vex controllers, and tether interface with video and audio jacks.

current flowing through it, depending on the motor direction. If either comparator detects an overcurrent condition, it pulls the wired-OR FAULT signal low which tells the controller to turn off the current to the motor in question until it drops to a safe level.

Each comparator input has a low-pass filter to measure average current. The time constant is three PWM cycles. This ensures that the average current of the motor is being measured, rather than any switching transients.

Controller Logic

The controller is implemented in a complex programmable logic device (CPLD). This device was chosen because it can be configured to perform multiple tasks simultaneously. A microcontroller executes only one program at a time, so it's very difficult to write software to decode a PWM input while simultaneously generating four PWM outputs.

A CPLD, on the other hand, can easily contain four motor control circuits, a couple servo output circuits, an input PWM capture module, and whatever else is needed. Since it doesn't execute a program, all these modules are active at all times.

The Xilinx XPLA3 series was chosen due to the author's familiarity with this device. The controller CPLD contents are described in the following section. The source code for the CPLD is written in VHDL — a hardware description language. It isn't a program to be executed line by line, but rather it is a functional description of Boolean logic implemented in configurable hardware.

The VHDL source code is available at the ESC4 website listed in the **sidebar**.

The Vex transmitter sends a six-channel pulse train with seven sequential pulses including a sync pulse, four motor speed pulses, and two on/off switch pulses. A 1.1 ms

long pulse signals full reverse, a 1.5 ms pulse is zero motion, and a 1.9 ms pulse is full forward. The sync pulse is 9 ms long. The above pulse times include the 400 μ s "off" time.

The pulses are decoded by the input module rcin.vhd.

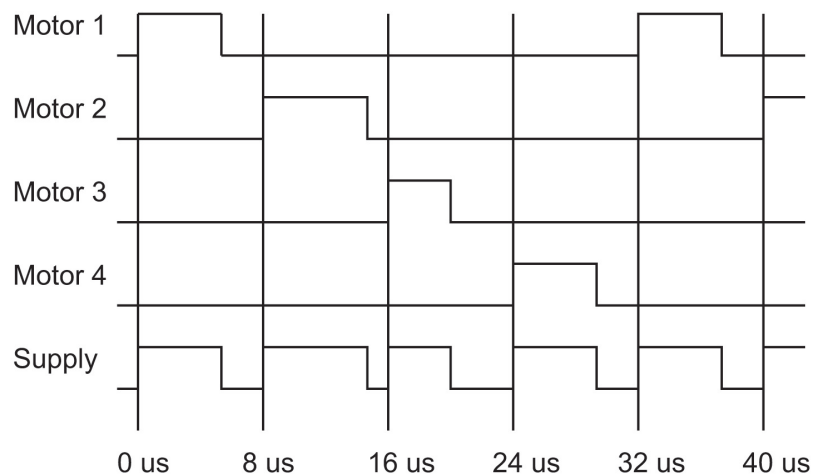


FIGURE 3.

The PWM current-sharing technique. Each motor uses one eighth microsecond time slot. The total supply current never exceeds that required for one motor.

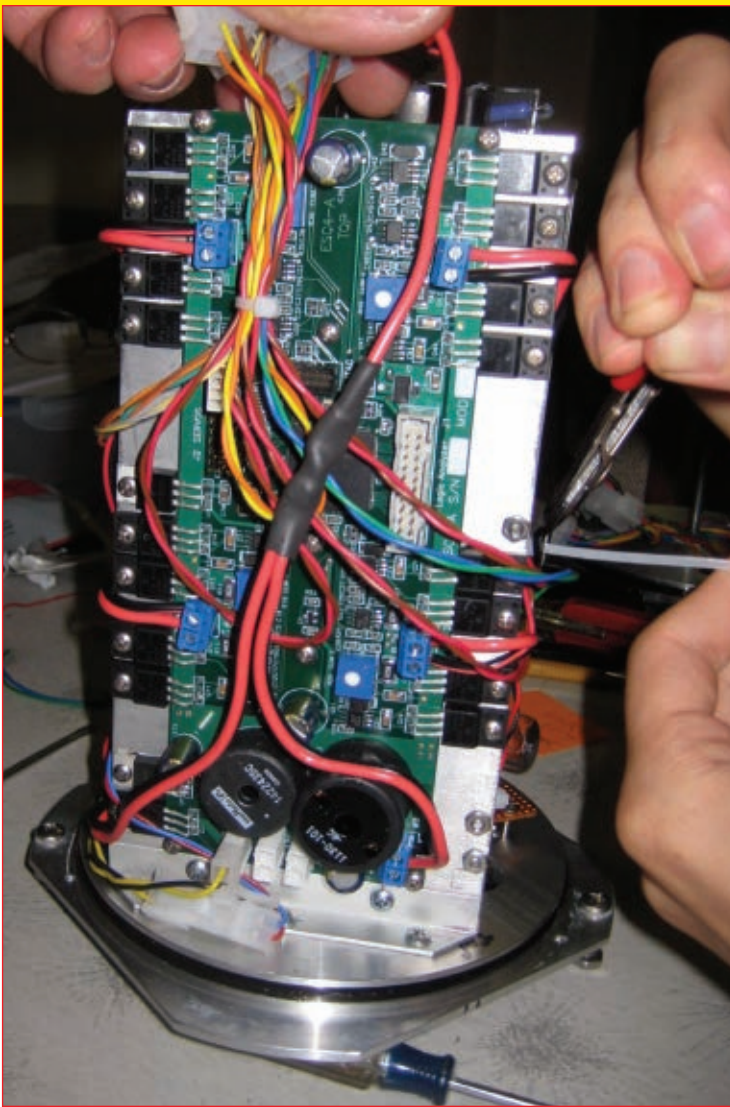


FIGURE 4.

The ESC4 quad speed controller circuit board.

direction bit steers the pulses to the proper H-bridge driver inputs. There's one dead clock cycle while switching the drivers over from active to inactive and back.

A PWM channel sequencer in `drive.vhd` issues sequential start commands to the four PWM engines in each eight microsecond time slice, allowing each motor to run for up to eight microseconds out of 32, for a 25% maximum duty cycle. The PWM downcounter runs at 25 MHz for up to 200 counts which corresponds to stick full forward or reverse.

The camera tilts with a standard RC servo. Since it's inconvenient to hold the spring-centered Vex joystick in the desired camera position, the joystick functions as a speed controller for the servo motion. This is done in the module `servo.vhd` by integrating the joystick command into a position accumulator that counts up or down at a rate proportional to the joystick command.

Each controller board has one switched 12V power output. This is implemented with an N-channel MOSFET on the low side. The load has + and - terminals that are both floating from ground. The control is one pair of on/off buttons on the rear of the Vex transmitter.

The speed and direction are presented to an internal data bus for distribution to the PWM and servo modules.

Each PWM motor controller in the module `pwm.vhd` converts an eight-bit speed command to a 31.25 kHz PWM stream whose duty cycle ranges from 0 to 25%. The

on/off buttons on the rear of the Vex transmitter.

Next Year

The basic design of notBob was sound, as evidenced by

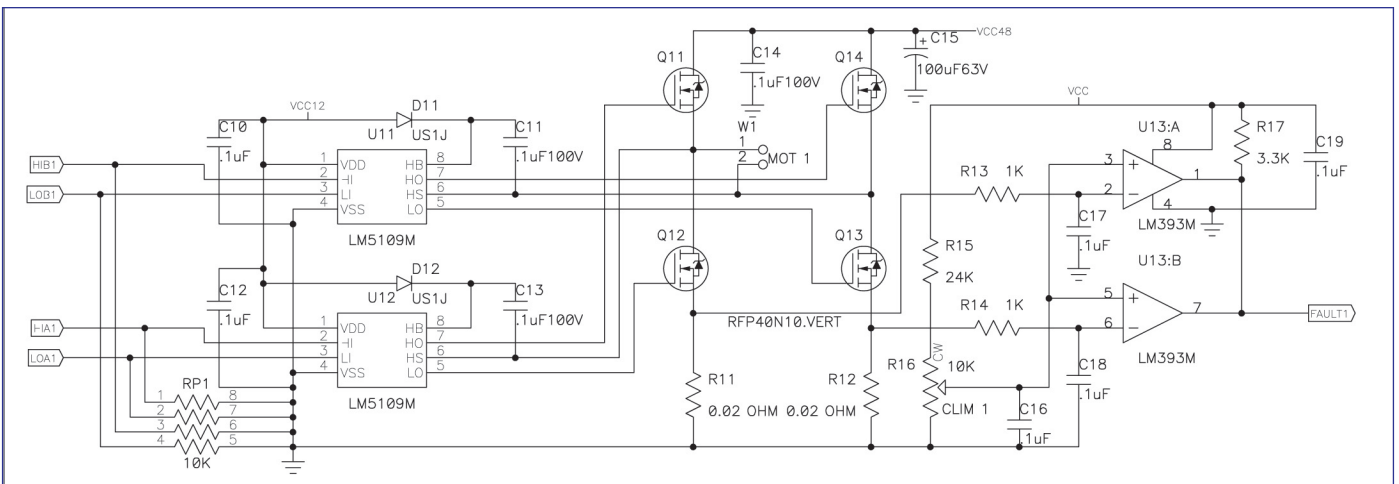


FIGURE 5. The schematic diagram of one PWM motor controller. The current limit circuitry detects overcurrent on either lower leg and instructs the CPLD to turn off the power momentarily.

the overall winner award we received at the 2009 NURC competition. The 2010 competition provides an opportunity to refine the design. More vertical thrust is high on the list of things to add. The new mission requires a more sophisticated manipulator. There are torpedoes!

The electronics are being optimized to do just what is needed with a bit of extra capability for unanticipated requirements. The big PC boards morph into a tiny card cage with a stack of speed controllers plugged into a backplane. A microcontroller decodes a serial command stream and returns telemetry over an RS-485 serial data link. The poolside Vex controllers are being replaced with a single Sony PlayStation 2 controller, using a field-configurable translator box with LCD display to format the commands. The mechanical design of the camera assembly is being simplified.

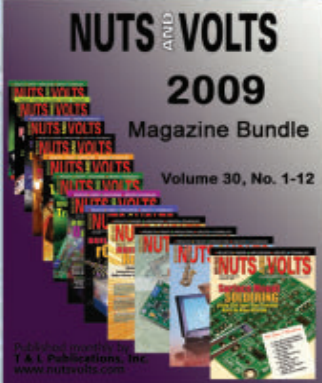
The old and new designs are accessible on our websites with schematics and Gerber files and parts lists for those who want to use them. **SV**

Further Reading

Speed controller board design details
www.cathodecorner.com/esc4/

The Typewriter Repairmen
2009 NURC
www.selectric.org/nurc/

National Underwater Robotics Challenge
www.h2orobots.org/

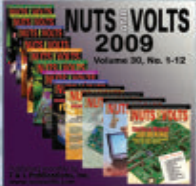


Choose your preferred format:
print or digital media.

We have just what you're looking for!

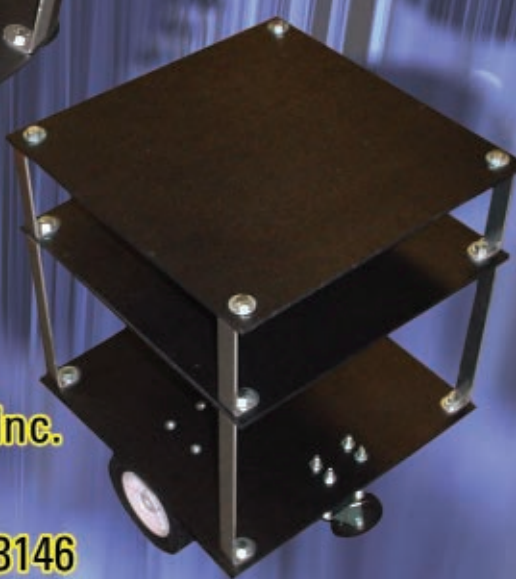
12 magazine bundle **\$57.00**
12 issues on CD-ROM **\$24.95**

Order today 800 783-4624
or online @ www.nutsvolts.com



Zagros Robotics

Large Mobile Robot Chassis Systems

Zagros Robotics, Inc.
P.O. Box 460342
Saint Louis, MO 63146

(314)341-1836
www.zagrosrobotics.com

Building Robot Parts with a Laser

By Jose Quinones

My brother and I often have this conversation about how machines will one day build machines. The conversation is highly philosophical and can easily get pointless. The scary aspect of it recently startled me when I realized I was already partially doing it when I was using a CNC Laser Engraver to build robot parts. That is, a robot is being used to build robots!





CNC stands for Computer Numeric Control. Equipment such as mills, lathes, drills, grinders, high pressure water jets, plasma cutters, routers (also referred to as rotary engravers), and laser engravers fall under this category. The tips and hints I will offer in this article can be adapted to any of the equipment previously mentioned. However, it is mostly intended for laser engravers or — like I will interchangeably call them — laser cutters.

Advantages and Disadvantages

A CNC laser is probably the most powerful tool I have ever operated. I am not talking about the power intrinsic to the amplified light which, in my case, is in the order of a mere 25W (1/4 of what my room lightbulb consumes). I am talking about the ability you acquire as a builder when operating one of these amazing power horses. I like to divide advantages into three categories:

1. **Accuracy:** Any CNC equipment enjoys the advantage of a computer controlling a set of actuators to get the desired job done. It is no secret that computer-controlled equipment can be considerably steadier than the human hand. As a result, you will see improved detail on laser cut parts than similar jobs done with a typical scroll saw.
2. **Repeatability.** So, you built your first robot. Awesome! How about building a second unit? Clearly, it can be done with hand-based tools, but CNC equipment gives you the opportunity of repeating the same process with the press of a button! Forget about setting up your saw, drill, lathe, and mill for how many times you'll need it. With a CNC laser, it is a matter of loading up the project file, placing the piece of material to cut inside of the laser equipment, and WOOSH! There it goes. In the same fashion, you could build hundreds

(or even thousands!) of robots and every time all you have to do is press the same button.

3. **Easy:** Using a CNC laser cannot be any easier. All you do is draw in the same vector fashion as you have drawn before with software packages like Corel Draw, Adobe Illustrator, or AutoCAD, and then print as with any other inkjet or laser printer. Although there are some parameters that need to be adjusted (if you are the laser operator), these are trivial when compared to parameter setup such as spindle speeds, blade materials, or the use of jigs and fixtures, etc. Like I said before: Load a file, place some material on the laser, and press go!

It all sounds like a pure breeze and no problems at all. Well, I would be dishonest if I were to tell you that there are only advantages and no disadvantages. There are some issues related with this technology, and we must overcome them as best we can.

CNC lasers are bulky and expensive. It would be awesome if everybody could afford one, but unfortunately that is not the case. A good CNC laser (plus all the other accessories such as air compressors and fume exhaust) will cost something in between \$10K and \$15K. Hence, mostly businesses tend to afford this kind of technology. Fortunately, there is a good deal of people out there trying to get their laser to pay for itself. They will gladly take your file and laser the artwork for a not-so-large fee.

Parts can only be 2D. With a CNC milling machine, three actuators move the cutting tool in the X, Y, and Z axis. With a CNC laser, however, the cutting takes place on the X and Y axis alone. There are ways to control the depth of the cut by modulating laser beam power, but the controllability of this depth is very poor as it depends on speed and material. Some CNC laser vendors will claim the tool to be 2.5D. I, on the other hand, look at the tool as a 2D tool and work around that. For example, you can always

Figure 1.



divide the job into layers and stack them up during the build time.

Not everything can be cut. Yes, you can process all sorts of materials on a milling machine, lathe, drill, saw, etc., but on the laser that is not the case. First of all, a CO₂ Laser can only cut materials such as woods and plastics (I am including acrylics under this category). To cut metal, you would need either a heinous amount of power or a totally different laser technology. Our robots are limited in terms of the materials they can be built from. No biggie! Most robots out there are made out of plastic, anyway!

Robot Bases

The best way to illustrate laser capabilities is to look at the most typical robot platform component: the base. You may want a circular, rectangular, triangular, or just some odd and complex polygon to shape up your project. Depending on the style selected, you would choose a tool to cut your material and get the best results. With a CNC laser, this is not the case.

To the CNC laser, a circle, a rectangle, or even a star are just a combination of line vectors to move about. Once commanded, the laser does what it does best — which is cutting by following the path formed by the respective vector array. Although it is clearly important to you, the path itself is irrelevant to the laser.

Figure 1 shows a circular base laser-cut from 1/4" MDF wood. I like

MDF because it is strong and very easy to work with. As long as my robot does not drive itself into a watery environment, it should last for a very long time. The main reason I like working with MDF is because it is reasonably cheap. Once I have experimented all I want and have obtained the part as I want it to be, I can then move into acrylic.

Experimenting with acrylic is rather prohibitive as it is considerably more expensive than MDF. Just to give you an example, I can go to my local Home Depot and buy a 24" x 18" acrylic pane for about \$20. However, a 24" x 48" MDF panel of the same width (1/4") is a little bit less than \$6. I then cut the MDF into three segments: two of them are 18" long; the third is about 12" long. So, for a third of the acrylic price, I get close to three times as much material.

Notice the base has a series of cavities and shapes I have worked into the material. Had I decided to cut this

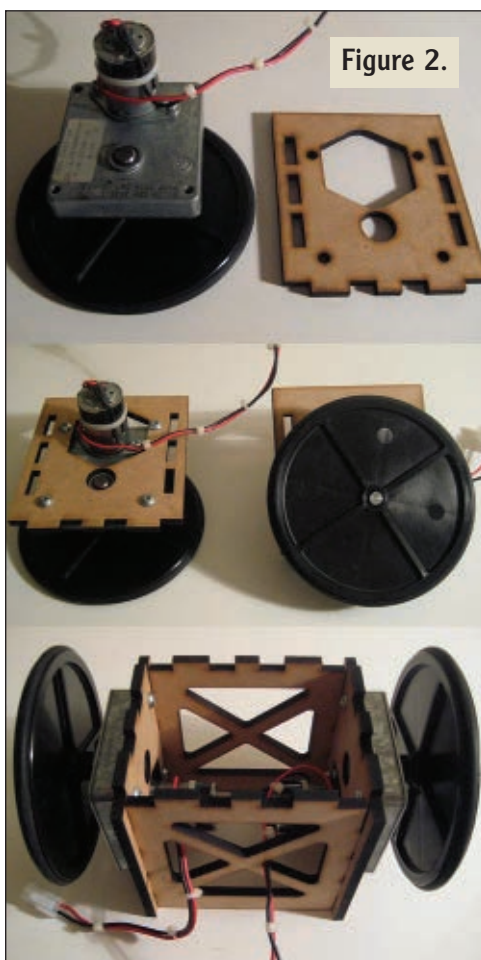
intricate shape with a series of power tools, it would have taken a great deal of time. Cut the circle first by utilizing some kind of rotary cut jig, and then using a scroll saw or router, cut the internal wheel holes. All of that effort would have taken hours (at least to me), but to the laser, it was around six minutes.

The base has holes which I will use to attach my standoffs to. To get these holes done, I could have simply used any drill. However, it is easier to have the laser cut the holes at the same time the base was being done. Cool thing is you can actually size these holes to practically any diameter. Whether the plan is to use 8-32, 4-40, or 10-32 screws, by specifying the diameter of the hole accordingly, that is precisely what we will get. A nifty trick I have found is to cut small enough holes so they can be taped later. No need for hex nuts!

Putting the Pieces Together

Since laser cutters are 2D machines, we need to get creative when we are trying to get our 3D

Figure 2.



creations together. There are three methods I can offer. They each have their own advantages and disadvantages.

Jigsaw Cut and Glue

Figure 2 shows a method to attach two motors together into an assembly which will then be attached to the robot base. Notice the series of mortise holes and aligning tenon extensions to the parts I plan on joining. This technique resembles a jigsaw puzzle and at the end, that's pretty much what it is. Make sure the mortise holes are the same width as the wood the tenon will be made of. I have made the classical mistake where the mortise is too small, and then the tenon will not go in.

The easiest thing to do with all of these parts is to glue them. This technique is easy and cheap. The bond is amazingly strong and the resulting contraption will practically last forever! However, I do not recommend this system for assembling robot platforms or components since once you have glued the parts together, disassembly is no longer feasible. The detail in **Figure 2** was not glued.

Standoffs and Screws

You could use metal standoffs if you happen to find the length you need. If you cannot find the right length, allow me to offer an alternative.

For standoffs, I am using wood dowels. The first step was to develop a quick jig such as the one you see in **Figure 3**. It allows the positioning of the large wood dowel and the cut of segments of different lengths. Because the large dowel sits at the same starting point at all times, we can direct the laser to the coordinate which defines the desired dowel length and make the cut.

Drill a hole through the dowel center with either a lathe or a bench drill. The resulting dowel can then be either tapped or you can add a threaded insert. I actually recommend the latter one. **Figure 4** shows the base assembled with wood dowels and screws.

Jigsaw and Screw

A hybrid between the two previous techniques would be the jigsaw and screw. In this method, the jigsaw portion is made such that it allows for a perpendicular screw and hex nut to be housed. This is trickier to draw, but it solves two problems at once. The box in **Figure 5** shows the technique in question. Notice the perpendicular surface has an opening to hold the hex nut while the first surface is made to hold the screw head.

Figure 3.

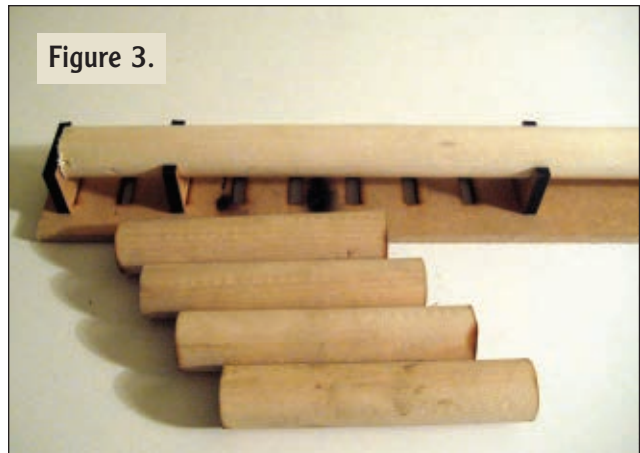
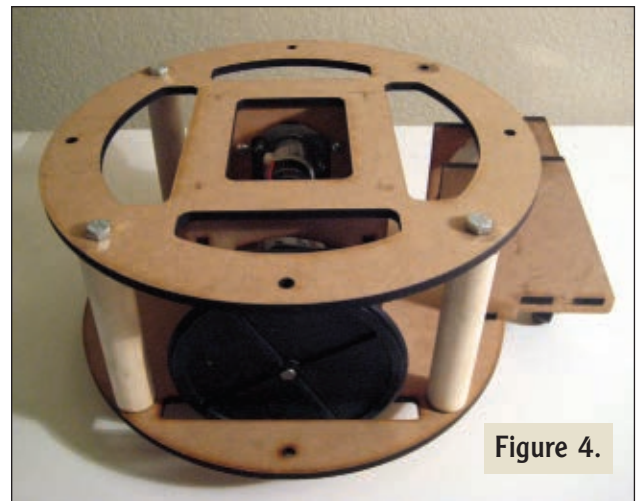


Figure 4.

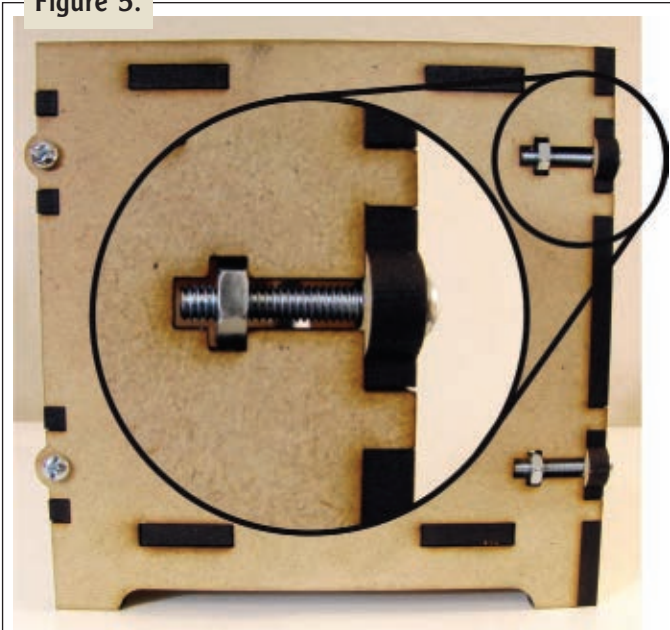


How Much Will All of This Cost Me?

Let's face it. There is no way I am going to exhort *SERVO* readers to go out and purchase a laser engraver. Although it is an awesome tool and you will have a blast with it, the cost of maintaining this piece of equipment is not a trivial endeavor. Luckily, you can easily rent laser engraving services and there are more than enough people out there waiting to laser your job out.

A good place to check out is **Pololu.com**. As you may know by now, Pololu is a great center to purchase your robotic parts, as well as kits, but as it turns out they also offer laser cutting services. They charge about \$2.50 per minute. The question may then arise as how do I know how many minutes will my job take?

Figure 5.



This is kind-of hard to say as there are a series of variables that must be taken into account. For example, a 25W laser will take longer to cut the same piece of material than a 50W or 100W laser. In the same fashion, the material width will play a role. If you want to laser-cut a piece with a 1/8" width, it will take less than the same piece but with a 1/4" width. Then comes the material itself. I have found that MDF is a little bit more time-consuming than say acrylic. There are other materials you can laser-cut such as lexan, ABS, and foam core, each of them taking different amounts of time to process.

The only way to get around this is to request a quote and compare rates. There are a great deal of laser shops out there and their rate tends to be in between \$60 to \$120 an hour. Each shop will have their own ways of charging for setup, design, and actual job rate, so feel free to ask around and stay with whoever gives you the best price.



Figure 6.

Drawing It into the Laser

If you are going to rent laser cutting services, I recommend that you draw your own artwork. You can submit all sorts of drawings and schematics, even on a napkin. However, if the laser shop operator has to spend time re-drawing the artwork and making it laser-ready, you will have to pay for it. Instead, get a few hours behind your computer and draw your part following these guidelines:

Use a vector-based program. Paint Shop, Photo Shop, or any one of those bitmaps software suites will not work. BMP files can be exported into vector-based SW such as Corel Draw or Adobe Illustrator but results are poor and lots of work needs to be done on the resulting artwork. You might as well just hand draw them on a napkin! You can also draw in AutoCAD and submit DXF or DWG files. They should be able to easily import these into their printing environment.

Define what you want to be cut versus what will be rastered. Laser engravers do one of two things: engrave (raster) or cut (vector). Whatever you want cut, draw it with a very thin line (either hairline or one point). Whatever you want to engrave — such as logos or cool images — draw with solids or lines that are four points or larger. The laser will instantaneously switch from vector cut to raster engrave depending on the information provided. **Figure 6** shows the same artwork but with two different line widths. The top version was laser cut, while the bottom version was raster engraved as can be seen on the right side.

Conclusion

The CNC laser is not only a powerful tool, it is also highly versatile. Thanks to its ability to carefully follow any intricate shape path, we can develop our own robot parts such as bases, panels, mounts, etc. Because it is computer controlled, it makes it very easy to build multiple parts of the same item with very little to no extra effort.

Although it is not a 3D CNC machine, there are ways in which we can be innovative and come up with our own ways to add depth to the projects at hand. The tool is available, all we have to do is get busy! **SV**

DESIGN THIS!

Linx Technologies, Inc., has announced an RF design contest jointly sponsored with Parallax, Inc. The contest is based on the new Parallax 433 transceiver board which incorporates the Linx LT Series transceiver. This easy-to-use and low-cost module is capable of sending and receiving serial data wirelessly between microcontrollers or to a PC, enabling the creation of applications limited only by the designer's imagination. There will be \$3,000 in cash prizes, as well as numerous product giveaways going to winning designs. Contest registration is underway and open until April 30, 2010. All entries must be received by May 31, 2010. Visit www.parallax.com/tabid/821/Default.aspx for complete details and contest rules.

Further information can be obtained by calling or visiting the respective websites at: www.linxtechnologies.com, (800) 736-6677; www.parallax.com, (888) 512-1024.

About Parallax

Founded in 1986, Parallax is a privately held company located in Rocklin, CA. Parallax designs and manufactures microcontroller development tools and small single-board computers that are used by electronic engineers, educational institutions, and hobbyists. The company's product line includes the popular BASIC Stamp® and Propeller Chip microcontrollers and development software, SX chips and programmers/debuggers, project boards, sensors, educational tools, robotics kits and accessories, and Propeller chips and tools.



About Linx Technologies

Incorporated in April 1997, Linx Technologies is a leading provider within the wireless industry of cost-effective, easily applied, and highly versatile radio frequency (RF) products. Today, Linx applies its capacity to simplify electronics to a range of products and technologies including: low-cost wireless voice, data, and GPS modules; USB interface modules; OEM RF products; and encoder/decoder ICs. Through its Antenna Factor and Connector City divisions, Linx also offers a wide range of standard and custom antennas, connectors, and cable assemblies.

NEW! Orangutan SVP-324 Robot Controller

The **Orangutan SVP-324** lets you program your robot in C/C++ with free AVR development software. Use Pololu's AVR C/C++ library to easily make your **motors** go, **servos** turn, **LCD** print, and **buzzer** play — the Orangutan has all of these hardware features (and more) built right in!

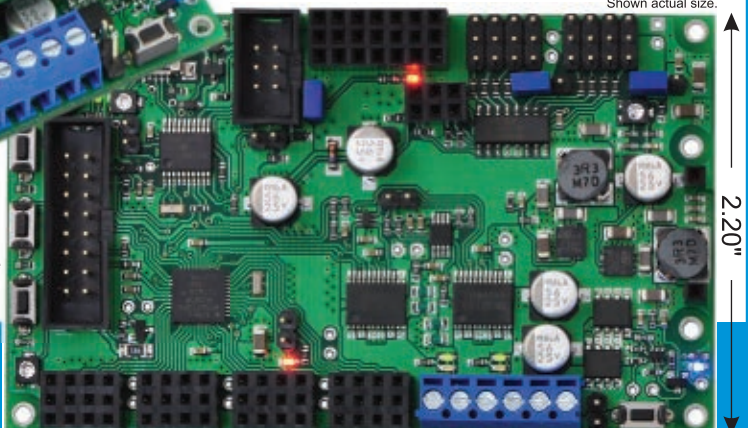
Pololu item #1325 & #1326: \$99.95 and \$89.95

The **Orangutan SVP-324** is available fully-assembled and as a kit.

Key Features:

- * **Wide input voltage range**—6–13.5 V.
- * **C/C++-programmable microcontroller**—User-programmable Atmel ATmega324PA AVR running at 20 MHz (32 KB flash, 2 KB SRAM, 1024 bytes EEPROM).
- * **Built-in USB AVR ISP programmer** (USB A to mini-B cable included).
- * **Two bidirectional motor ports**—2 A continuous/6 A maximum per channel.
- * **8 servo ports**—Use Pololu's AVR C/C++ library to easily control up to 8 servos!
- * **21 free I/O lines**—Up to 12 can be analog inputs, optional dual quadrature encoder inputs, two hardware UARTs.
- * **Removable LCD**—16-character × 2-line, with backlight.
- * **Two step-down voltage regulators**—Each capable of supplying 3 A.
- * **Other fun stuff**—Buzzer, 3 user pushbuttons, 2 user LEDs, and more!

Shown actual size.



Pololu
Robotics & Electronics

3095 E. Patrick Ln. #12, Las Vegas, NV 89120

Save 10%
With coupon code
SVP324SRV3

Learn more at www.pololu.com/svp or by calling 1-877-7-POLOLU.

The *SERVO* Webstore

Attention Subscribers ask about your discount on prices marked with an *

CD-ROM SPECIALS



5 CD-ROMs & Hat Special
Only \$ 109.00 (includes shipping)!

www.servomagazine.com



On Sale Now!
Only \$24.95

ROBOTICS

PIC Robotics

by John Iovine

Here's everything the robotics hobbyist needs to harness the power of the PICMicro MCU!

In this heavily-illustrated resource, author John Iovine provides plans and complete parts lists for 11 easy-to-build robots each with a PICMicro "brain." The expertly written coverage of the PIC Basic Computer makes programming a snap – and lots of fun.

\$24.95



Forbidden LEGO

by Ulrik Pilegaard / Mike Dooley

Build the Models Your Parents Warned You Against.

Forbidden LEGO introduces you to the type of free-style building that LEGOs master builders do for fun in the back room. Using LEGO bricks in combination with common household materials (from rubber bands and glue to plastic spoons and ping-pong balls) along with some very unorthodox building techniques, you'll learn to create working models that LEGO would never endorse. **\$24.95**

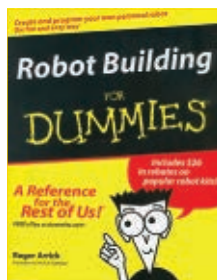


Robot Building for Dummies

by Roger Arrick / Nancy Stevenson

Discover what robots can do and how they work. Find out how to build your own robot and program it to perform tasks. Ready to enter the robot world? This book is your passport! It walks you through building your very own little metal assistant from a kit, dressing it up, giving it a brain, programming it to do things, even making it talk. Along the way, you'll gather some tidbits about robot history, enthusiasts' groups, and more.

\$24.95



Build Your Own Humanoid Robots

by Karl Williams

GREAT 'DROIDS, INDEED!

This unique guide to sophisticated robotics projects brings humanoid robot construction home to the hobbyist. Written by a well-known figure in the robotics community, *Build Your Own Humanoid Robots* provides step-by-step directions for six exciting projects, each costing less than \$300. Together, they form the essential ingredients for making your own humanoid robot. **\$24.95***



Robot Programmer's Bonanza

by

John Blankenship,
Samuel Mishal

The first hands-on programming guide for today's robot hobbyist!

Get ready to reach into your programming toolbox and control a robot like never before! *Robot Programmer's Bonanza* is the one-stop guide for everyone from robot novices to advanced hobbyists who are ready to go beyond just building robots and start programming them to perform useful tasks.

\$29.95



Robotics Demystified

by Edwin Wise

YOU DON'T NEED ARTIFICIAL INTELLIGENCE TO LEARN ROBOTICS! Now anyone with an interest in robotics can gain a deeper understanding – without formal training, unlimited time, or a genius IQ. In *Robotics Demystified*, expert robot builder and author Edwin Wise provides an effective and totally painless way to learn about the technologies used to build robots! **\$19.95**



We accept VISA, MC, AMEX,
and DISCOVER
Prices do not include shipping and
may be subject to change.

To order call 1-800-783-4624

SERVO Magazine Bundles



Published by T & L Publications, Inc.

\$57
per bundle

Save \$10
off the
normal
price!!

Now you can get one year's worth of all your favorite articles from *SERVO* Magazine in a convenient bundle of print copies. Available for years 04, 05, 06, 07, and 08.

Kickin' Bot

by Grant Imahara

Enter the arena of the metal gladiators!

Do you have what it takes to build a battle-ready robot? You do now! Here are the plans, step-by-step directions, and expert advice that will put you in competition — while you have a heck of a lot of fun getting there. Grant Imahara, the creator of the popular BattleBot Deadblow, shares everything he's learned about robot design, tools, and techniques for metal working and the parts you need and where to get them.

\$24.95

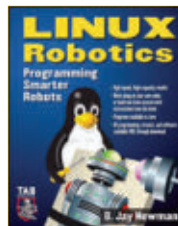


Linux Robotics

by D. Jay Newman

If you want your robot to have more brains than microcontrollers can deliver — if you want a truly intelligent, high-capability robot — everything you need is right here. *Linux Robotics* gives you step-by-step directions for "Zeppo," a super-smart, single-board-powered robot that can be built by any hobbyist. You also get complete instructions for incorporating Linux single boards into your own unique robotic designs. No programming experience is required. This book includes access to all the downloadable programs you need.

\$34.95

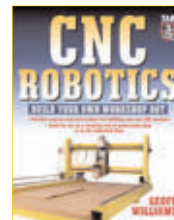


CNC Robotics

by Geoff Williams

Here's the FIRST book to offer step-by-step guidelines that walk the reader through the entire process a building a CNC (Computer Numerical Control) machine from start to finish. Using inexpensive, off-the-shelf parts, readers can build CNC machines with true industrial shop applications such as machining, routing, and cutting—at a fraction of what it would cost to purchase one. Great for anyone who wants to automate a task in their home shop or small business

\$34.95



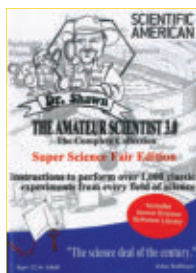
SPECIAL OFFERS

The Amateur Scientist 3.0 The Complete Collection

by Bright Science, LLC

There are 1,000 projects on this CD, not to mention the additional technical info and bonus features. It doesn't matter if you're a complete novice looking to do their first science fair project or a super tech-head gadget freak; there are enough projects on the single CD-ROM to keep you and 50 of your friends busy for a lifetime!

Reg \$26.95 Sale Price \$23.95



Getting Started Combo



The Getting Started Combo includes: Getting Started in Electronics by Author Forrest Mims and the DIY Electronics Kit. In his book, Mims teaches you the basics and takes you on a tour of analog and digital components. He explains how they work and shows you how they can be combined for various applications. The DIY Electronics Kit allows for the hands-on experience of putting circuits together -- the kit has over 130 parts! No soldering is required and it includes its own 32 page illustrated manual.

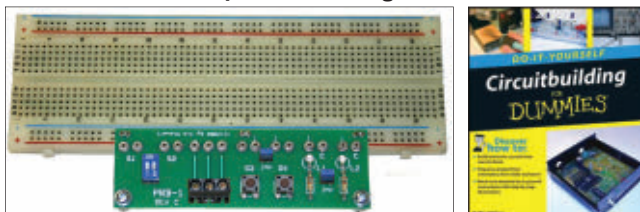
Combo Price \$62.95 Plus S/H

SPECIAL OFFERS

Proto Buddy Kit & Book Combo

For those just getting started in electronics as a hobby, a solderless breadboard (SBB) is the perfect platform for building those first circuits. Attach a Proto Buddy to an SBB, include a battery or two, and you will have a combo that has a lot of the same functionalities as more expensive units.

Combo includes PCB & Components, 830 point SBB, and Do-It-Yourself Circuitbuilding For Dummies.

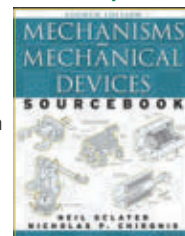


Combo Price \$57.95 Plus S/H
Limited time offer.

Mechanisms and Mechanical Devices Sourcebook,

by Neil Sclater,
Nicholas Chironis

Over 2,000 drawings make this sourcebook a gold mine of information for learning and innovating in mechanical design. Overviews of robotics, rapid prototyping, MEMS, and nanotechnology will get you up-to-speed on these cutting-edge technologies. Easy-to-read tutorial chapters on the basics of mechanisms and motion control will introduce those subjects to you. **Reg \$89.95 Sale Price \$69.95**



Getting Started With PICs Volume 2

by Chuck Hellebuyck

Getting Started With PICs Vol 2 continues where the first book left off. This book contains a collection of the next 12 articles that were published in *Nuts & Volts* magazines of 2007. A great complement to Vol 1 and a must have if you're entering the world of PICs and Programming.



Reg \$29.95
Introductory Price \$24.95

Enter into the world of
PICs & Programming
with this great
combo!

Combo Price \$155.00

For complete details visit our webstore @ www.servomagazine.com

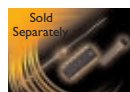
PROJECTS

Das Blinkenboard Kit

As seen in *Nuts & Volts* June issue,
Personal Robotics by Vern Graner.

This kit includes a pre-programmed ATtiny84 microcontroller that sports eight software PWM channels to control motor speed and light brightness. Jumper selectable patterns can be used to operate motors, solenoid valves, relays, or any DC load up to 24V/500 mA per channel!

Expand your board with the
"Der Magnetfelder
Detektor" componet pack.



Subscriber's Price \$32.45
Non-Subscriber's Price \$35.95
PCBs can be bought separately.

16-Bit Micro Experimenter Board

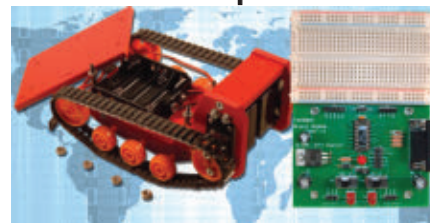


Ready to move on from eight-bit to 16-bit microcontrollers? Well, you're in luck! In the December 2009 *Nuts & Volts* issue, we will introduce you to a new, "16-bit Micro Experimenter."

The kit comes with a CD-ROM that contains details on assembly, operation, as well as an assortment of ready-made applications. New applications will be added in coming months.

Subscriber's Price \$55.95
Non-Subscriber's Price \$59.95

Tankbot Kit & Brain Alpha Kit



As seen in the Sept. issue
Tankbot/ Brain Alpha
by Ron Hackett

A series filled with projects and experiments to challenge you through your learning process while you grow your fully expandable Brain Alpha PCB!

The brain is a PICAXE-14A!

For more info & pictures, visit the
SERVO Webstore.

Tankbot and the Brain Alpha Kit
can be purchased separately.

Combo Price \$ 138.95

$$Velocity = \frac{Acceleration * Time}{velocity_2} + initial$$

$$P = T * \omega$$

$$P_{hp} = 6.72 \text{ watts} * \frac{1 \text{ hp}}{746 \text{ watts}} = .009 \text{ hp}$$

$$(2 * \pi * Drive wheel$$

$$M = \frac{T}{(a + \sin \theta) * r}$$

Base on a Budget

By Chris Savage

Sometimes building your own robot is easier said than done. It can be easy to visualize what we want but then to actually *build it* can be more difficult or, in some cases, seemingly impossible. Other times, it is more a matter of not having a budget for the entire robot. Spare parts can be found in a number of places and different people have different interests in specific microcontrollers so I won't focus on that in this article. Instead, the focus here will be on getting a useable robot platform built on a minimal budget.

FIGURE 1.



Parallax
motor mount
and wheel kit.

The Drive Train

The drive train determines how the robot will move so that is the first thing you should decide on before getting started. Wheel robots are the easiest to build and often the simplest to control in terms of movement, so that is what will be used in this article as an example. Parallax, Inc., offers a motor mount and wheel kit which I'll be using in this article (sans wheel encoders), but you can use any type of drive motor and wheel combination to achieve the same results as you will see in some of the examples.

A Wooden Robot?

Why not? When we think of robots, most of us probably wouldn't think of wood. We would visualize a metal robot or even plastic. However, wood has several

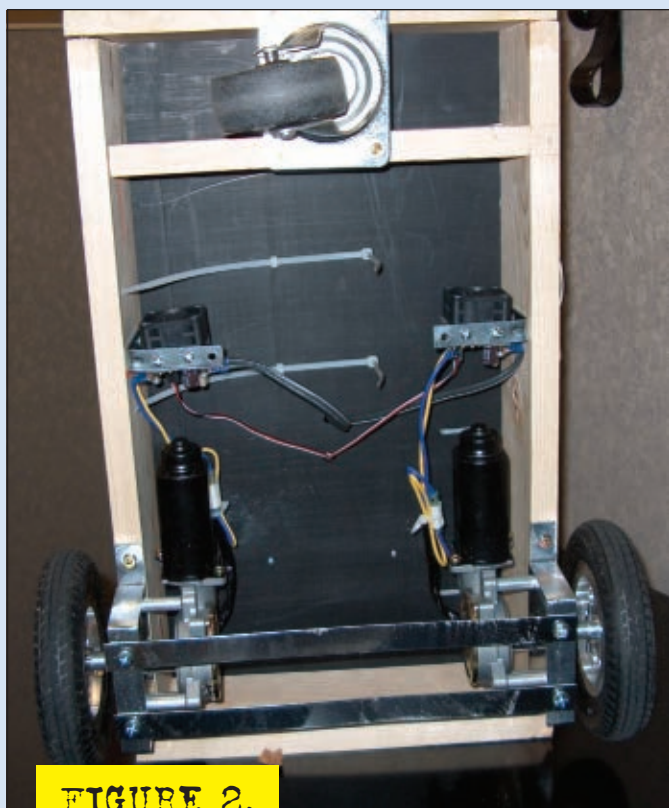
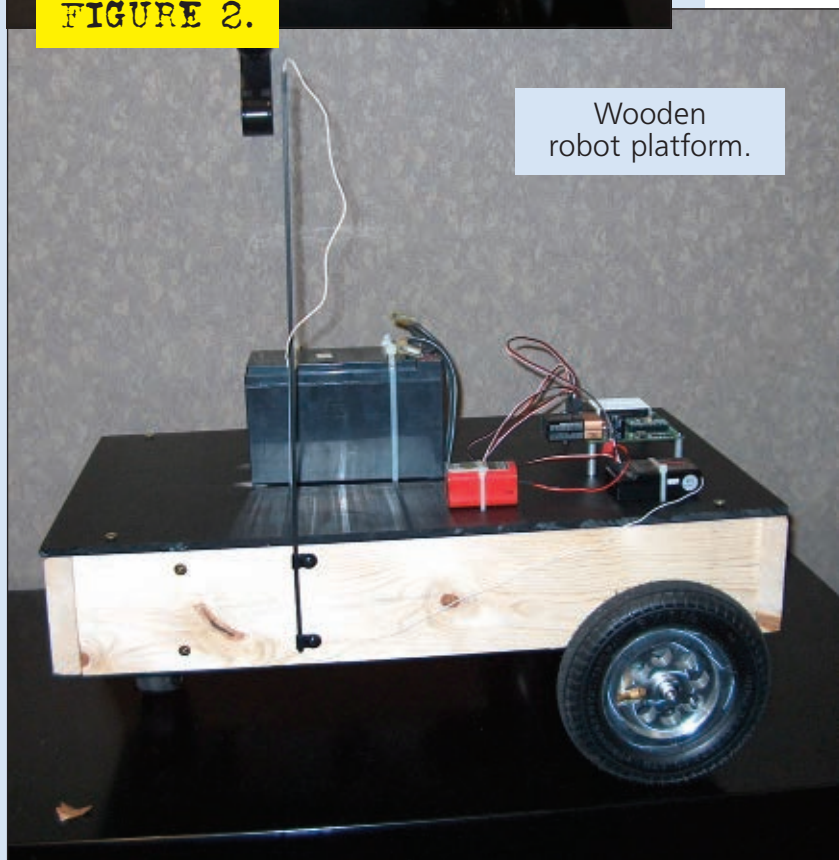


FIGURE 2.



Wooden
robot platform.

advantages over metal and plastic for the hobbyist on a budget. For starters, it is cheap. You can get wood for a robot from any hardware store and an inexpensive saw to cut it if you don't have one. It is durable and also lends itself to using wood screws instead of nuts and bolts to secure parts. This can further reduce the cost of mounting hardware. It is also easy to cut into the shapes you need without requiring special tools or a mold.

The robot in **Figure 2** is built from several pieces of wood shaped into a kind of box. Everything was held together using sheetrock screws and the accessories (such as the wheels) were attached using wood screws. The total cost of the wooden chassis including hardware and tail wheel was \$10. The wood was pine, the tail wheel was a generic caster wheel, and the hardware was a handful of screws all purchased from a local hardware store. Two Parallax HB-25 motor controllers were being used to control the motors in this robot. Power and signal wires are routed through holes in the top to a BASIC Stamp 2 on a Board of Education. A sealed lead-acid battery was salvaged from a radio-controlled tank project and used for power, and for testing a radio was salvaged from a crashed R/C plane to allow for selective radio control of the robot.

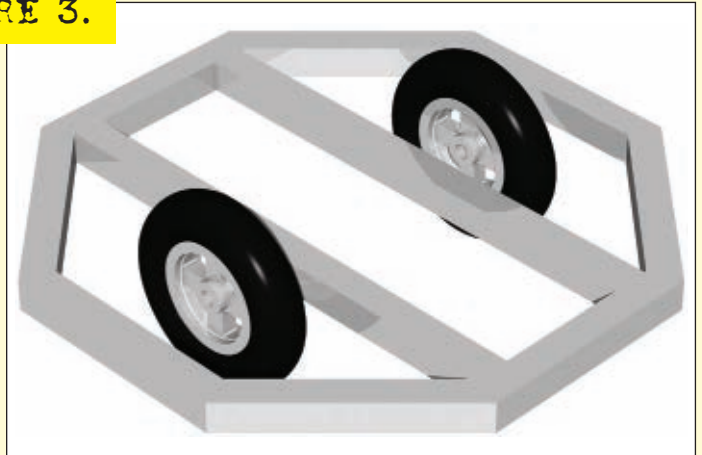
Moving Up to Aluminum

When I finished the wooden robot chassis, I posted it to a thread in the Parallax support/discussion forums where members were talking about creating a robot chassis for the wheel kit (which at that time had not been released yet). The thread is 17 pages long and a link to it is listed in the **Resources sidebar** if you are interested in the discussion that led to the robots discussed in this article.

Some of the feedback was related to creating a metal chassis and I had already considered what I thought to be the ideal platform in my mind, though I had not built it yet. Parallax forum member MarkS rendered my description into **Figure 3**. While this was what I wanted to build, I too was constrained by both a budget, as well as the need to create something easy to build. I finally decided on a square frame and the results can be seen in **Figure 4**.

This robot chassis was built using a single piece of 1" aluminum square tube

FIGURE 3.



Forum member's concept of aluminum chassis.

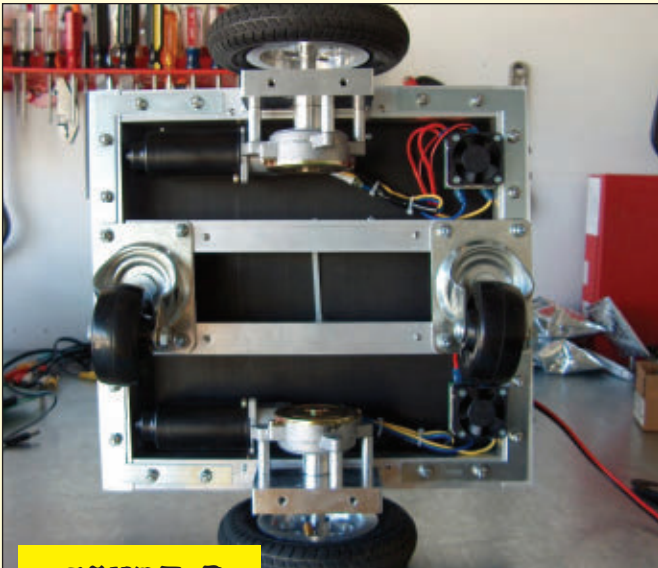


FIGURE 5.

Details of HB-25 and caster mounting.

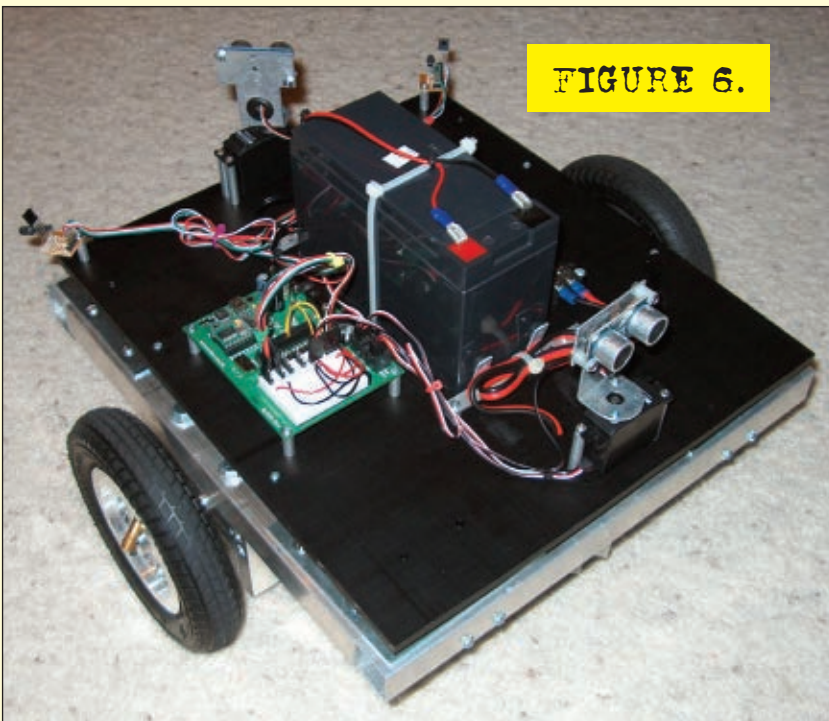


FIGURE 6.

Completed robot with sensors.

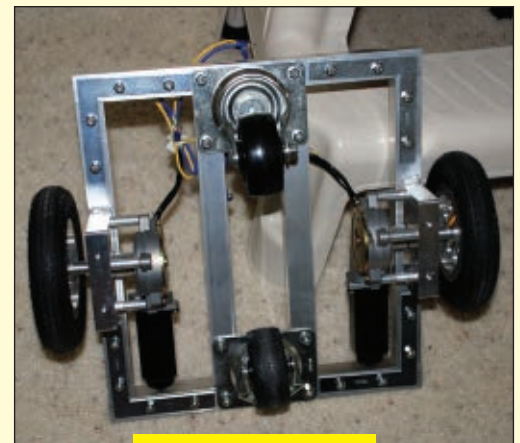
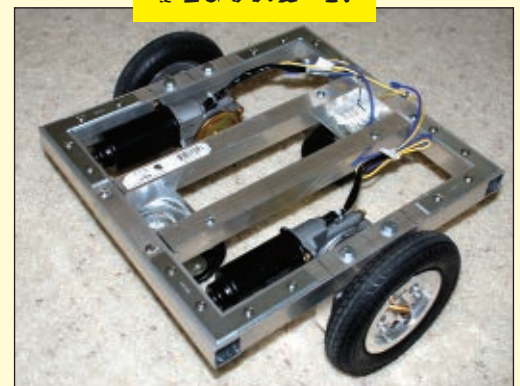


FIGURE 4.



Square aluminum chassis.



FIGURE 7.



Big BoE and bigger brother.



FIGURE 9.



Plywood robot by Ken Gracey.

(1/20" thick) obtained from a local hardware store. The tube was cut into four pieces measuring 11-1/2" long and two pieces 12-1/2" long. The pieces were held together using right angle brackets and some screws. The only tools used were a hacksaw and a drill. The hardware consisted of a handful of nuts, bolts, and washers. This time, two caster wheels were used bringing the cost of this chassis to just \$20! Still very modest, and the same type of motors and wheels were added, but again, any surplus units will work as well.

From Wood to Metal and Back Again

One of the great things about a tight-knit robotics community is that you always have a large group of dedicated hobbyists willing to share their thoughts and ideas. Many post pictures of their completed work and some of the ideas that were spawned in the aforementioned thread include the following robots. In **Figure 7** is Big BoE and his even bigger brother by Brian_B. In **Figure 8**, Ken Gracey of Parallax shows how even plywood can be used as a robot base. This robot gained much interest, and forum member Tom C created his own version and later added a second deck. Tom's achievement is shown in **Figure 9**. One thing you may have noticed about the two-deck version is that Tom used threaded rod for the standoff/spacers. This proves that you really can find a lot of good solutions at your hardware store.

Build Your Own Wooden Robot

The plywood robot in **Figure 8** was a concept robot that Ken Gracey built to show how useful plywood could be in making a robot chassis. Ken recently refined his wooden platform using additional Parallax hardware to accent the wooden base. He even added CCFL tubes and special machined PING mounting brackets. The revised design is shown in **Figure 10**. By request of the Parallax forum community, Ken is making the DXF drawing of this robot base available for free to everyone. You can download the DXF file from the *SERVO Magazine* website (www.servomagazine.com) or from the project resources link in the **sidebar**.

Final Thoughts and Additional Resources

Robotics can be expensive or it can be done on a budget. I have seen robot bases made from balsa wood, plastic pill bottles (solar bots), a mailbox, a toaster, a wheel chair, and even from recycled toys such as radio-controlled tanks and cars. Once you have the base, adding the electronics is the next step and varies based on your microcontroller platform.

Examples of other robot platforms can be found on the Savage Circuits main project site. Complete details of the aluminum chassis in this article can be found at the project page in the **Resources sidebar**, as well as a link to the discussion forum for the project. Challenge yourself to make something out of recycled parts and don't be afraid to add accessories to your robot that are recycled from other toys. One common accessory to recycle from toys and add to robots is the Nerf® missile launchers.

In a future article, I will be discussing optical wheel encoders for robots. While I will be using the Parallax Stingray robot platform as a base for the experiments, these concepts can apply to any wheeled or tracked robot and are an important step to your robot knowing how far it travels, moving in a straight line, and making consistent calibrated turns. **SV**

Resources

Project Page
www.savagecircuits.com/wheelbot/

Project Discussion
<http://forums.savagecircuits.com/index.php?topic=38.0>

Parts Resources
Parallax, Inc.
www.parallax.com

Robot Discussion Thread:
<http://forums.parallax.com/forums/default.aspx?f=10&p=001&m=195044>

RadioShack
www.radioshack.com

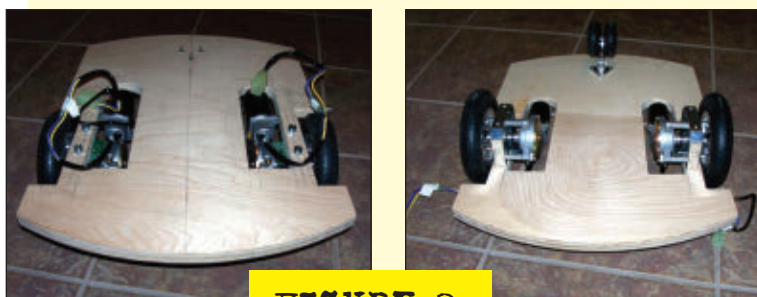
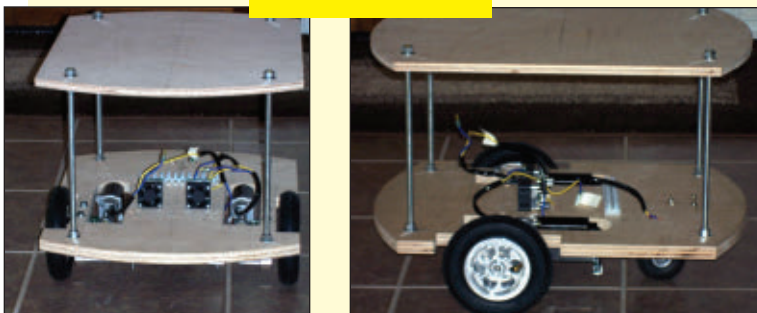


FIGURE 9.



Tom's double-decker wooden robot chassis.

FIGURE 10.



Ken Gracey's revised wooden platform.

Teach Your Old RoboScout Some New Scripts

By Robert Doerr



A while back, Sharper Image sold a personal robot called the RoboScout. It is a remotely controlled robot that also has some built-in routines to roam around the house, act as a security guard, and even welcome guests that may come over. The design of the robot is really unique and it is a fun robot to own. RoboScout is a simple robot to use and one that the kids enjoy. It is one that I often bring along when performing robot demonstrations. Unfortunately by default, there was no easy way to reprogram the robot or make it do anything other than what came with it.

For a short period, Sharper Image did sell an optional SI983 control box that connected to your PC with a USB cable. This PC interface would allow your PC to act as a replacement for the RF remote. A virtual remote control panel was displayed on the screen for controlling the robot. This feature is extremely handy if the original remote is ever lost since it can do everything the physical remote could. Included in the option kit was a PCI video capture card so the video feed from the robot could be seen on the PC to fully emulate the remote. Besides acting as a virtual remote, it was supposed to allow for programming the robot but there wasn't much documentation or tutorials to explain how it worked. Here was this great little interface but no one really knew how to use it. Well, this article will shed some light on that and help you use the optional PC interface to write scripts that enable RobotScout to accomplish some new tricks.

Before we get started on writing these scripts to program the robot, let's go over some details on the robot itself, the use of the remote, and the optional PC interface.



The most incredibly helpful personal robot!

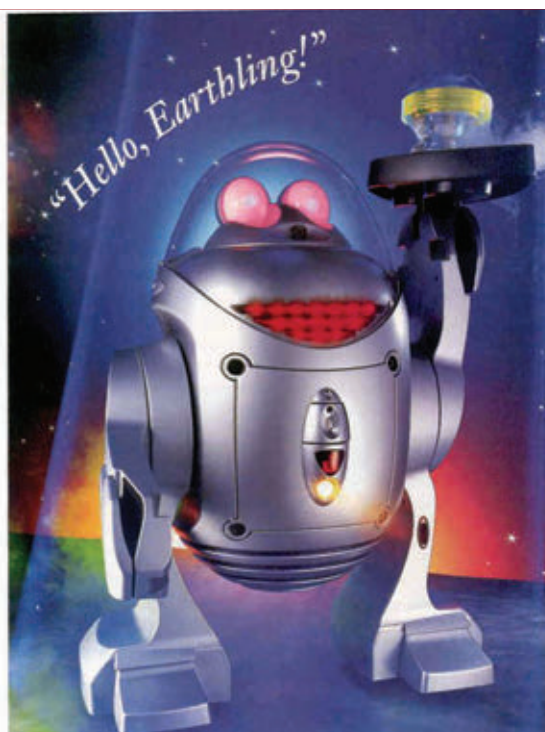
Imagine — a walking, talking, seeing, hearing personal robot! RoboScout® has large, dynamic eyes that truly “see” through a hidden-eye surveillance camera. Its 2.4 GHz transmitter sends clear images to a 3- x 4-inch LCD on a wireless remote-control center. Hidden microphones pick up sounds and allow conversations back and forth, too. RoboScout has a large vocabulary — greetings (“Hello, bello, bello!”), exclamations (“Watch out!”), plus ditties, commands, protests and questions. Select special modes: In “SENTRY,” RoboScout’s sensors detect motion and trigger voice alerts. Switch to “MONITOR” to keep an eye on a child at play. Press “GREET” and RoboScout will welcome guests. Press “MESSAGE” to record a 15-second memo. Both the RoboScout and its controller have built-in rechargeable batteries; each comes with a charging adapter. RoboScout stands 25 inches tall. The remote is 12 inches wide; its LCD is 3 x 4 inches. Optional USB Controller puts all functions on your PC! (SI983, \$199.95 if purchased separately). One-year warranty. Created by Sharper Image Design.



Wireless 900 MHz remote.



RoboScout® Personal Robot
SI583 (without PC interface) \$399.95



The original RoboScout advertisement from 2003.

Overview of Robot and Remote

The Sharper Image RoboScout (catalog #SI583) stands 25 inches tall and weighs about 24 lbs. It can move forward, back, right, and left manually via the remote or by itself in Auto-Roam mode. Sensors detect obstacles and stairs when running autonomously. Two adjustable arms hold up to two lbs of drinks and snacks on two trays included with the robot. The remote can transmit the operator’s voice through the RoboScout in your voice or alter it for a custom robot voice. You can also see what the robot sees since the 2.4 GHz transmitter sends video/sound back to the remote’s LCD screen. Rechargeable NiCad battery packs are standard in the robot and remote. You’ll know how much power is left on the robot since it has an LED bar graph as an indicator on the back by the charging plug.

When you power-up the RoboScout, it will run through some self tests and reset the positions of the arms, head, etc., to the home position. Once it is done, you are ready to use the remote

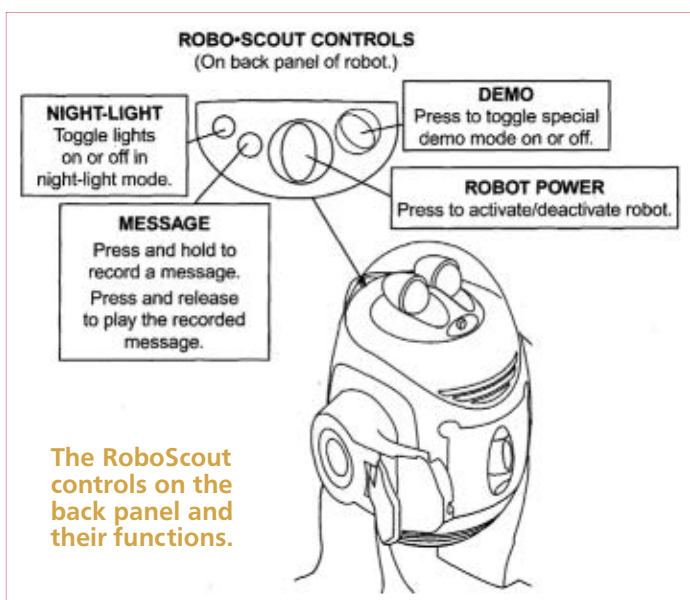
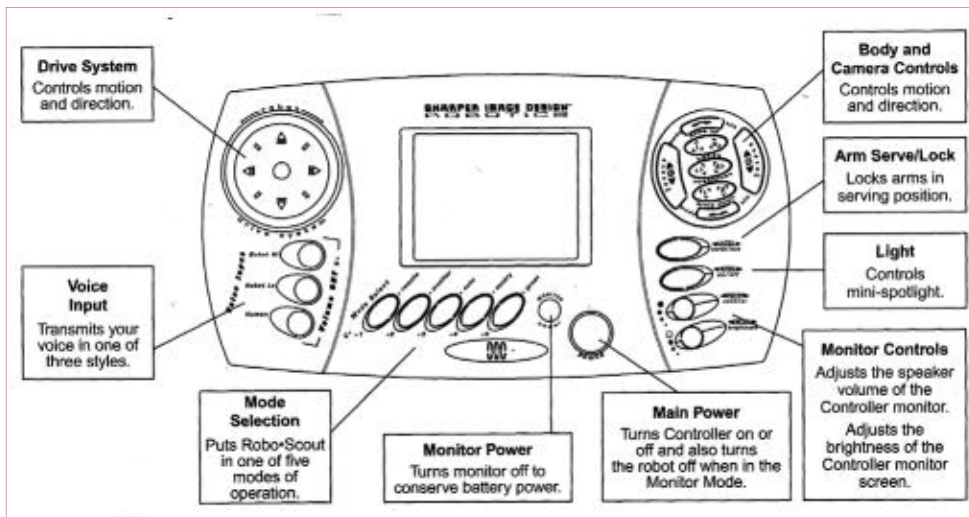
control or run scripts with the PC interface. At this point, you can also run the built-in demo by pressing the DEMO key on the back of the robot. Turn the night light on/off or use the built-in message recorder to leave a message for a family member or friend.



Using the Remote Control

The remote control for the RoboScout is pretty impressive. It is a huge remote that is packed with many features. It has navigation buttons on the left that allow you to control the movement of the robot and manually drive it around. In the center, there is a large 3” x 4” LCD screen which displays what the robot can currently see with its camera. To change the camera angle, on the right side of the remote you’ll find body and camera controls (eye camera left and right) and body tilt up and down so you can get a better view. Even when not using the camera, these movements help give the robot character for everyone watching it. There are also several extra buttons for moving the arms, hands, changing the state of

The RoboScout remote with its features documented.



The RoboScout controls on the back panel and their functions.

the spotlight, projecting your voice through the robot, and changing modes. Once you are familiar with the operation of the remote, you can manually run through the routine you want to get a feel for it. As you go along, jot down each step and use them as the basis for your own scripts.

There are five selectable operating modes of the robot that you can select using the remote control or virtual remote with the PC interface. These are:

Remote: In remote mode, the controller operates all movements as well as the voice and camera. This mode is also used to adjust the camera before switching to monitor mode.

Monitor: In monitor mode, the camera and microphone on RoboScout are both active. Press power on the controller to switch both the monitor and RoboScout to standby.

Auto-Roam: RoboScout moves around by itself and stops to greet people it encounters. Use the Arm-Lock feature with optional trays to have RoboScout serve refreshments.

Sentry: Detects intruders and sounds the alarm/alert. Once activated, you have 30 seconds to evacuate the area before the detector sweeps the room for sound and motion.

Greet: Greets people as they approach in a manner similar to the Auto-Roam mode. In greet mode, RoboScout does not move about the area.

Using the Optional PC Interface

To do custom programming of the RoboScout, you'll need to get the RoboScout PC Interface SI983 option. The brain of the control box is a Motorola 908JB8 microprocessor. This microprocessor acts mainly as a gateway between the RF link to the robot and the USB link to the PC. This enables the host PC to control the robot and emulate the RoboScout remote. There are two connections from the PC interface to the host computer. One is the USB



The RoboScout PC interface (A); connections to the PC interface (B); and the PCI video capture card included with the PC interface (C).

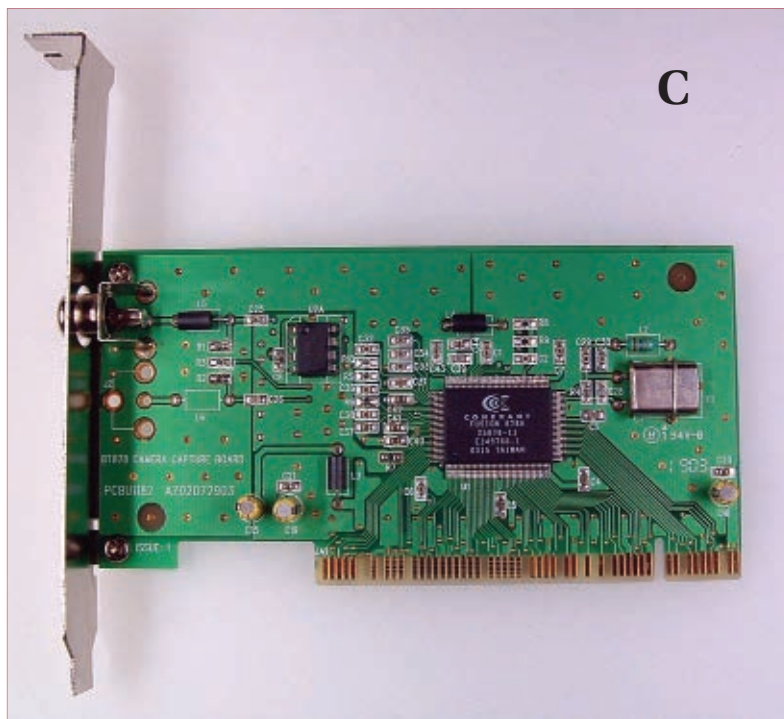
interface for controlling the robot. The other is the audio/video cable which is used to receive the video from the camera in the robot, and also for the sound out from the host computer which is projected through to the robot.

All of the RoboScout control programs are only available for the Windows platform. The original v1.0 software that comes with the RoboScout interface required the PCI video capture card to be installed in the host computer or the software won't run. It was a silly constraint since it is only used to display the robot's camera on the virtual remote screen. A patched version of the control software v1.1 was later released that allows you to select a check box during the install which disables the video capture. This will allow the software to run without the PCI capture card so it can be used with laptop computers. The video signal from the robot is just a standard composite signal which could be easily displayed by other methods.

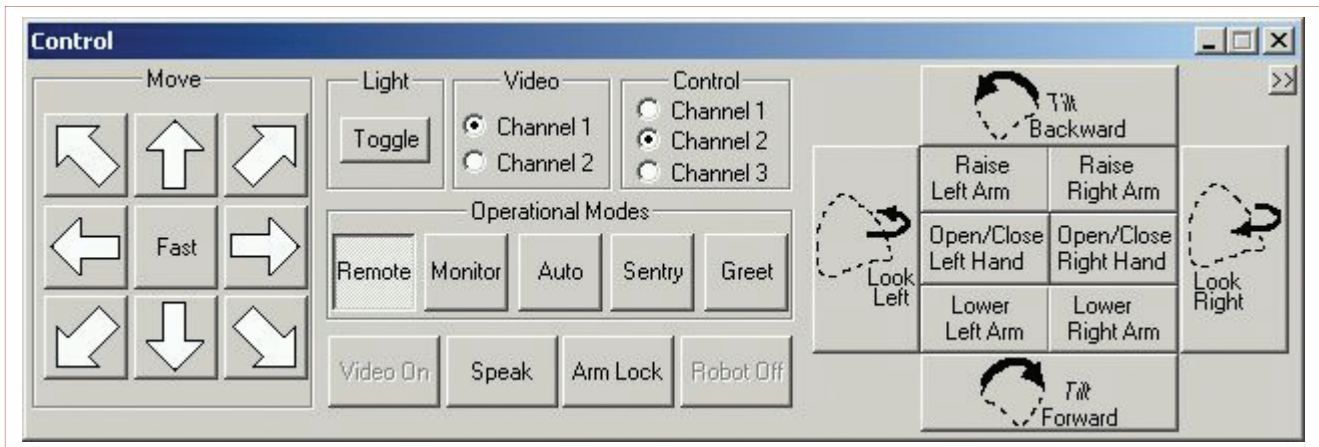
If you are using a desktop PC, you can connect the PC interface directly to an available USB port on your computer. Although most desktop PCs may not need external power, I did notice an improvement when using a laptop to drive the PC control box. Without it, some of the speech would not come through and some of the commands were skipped and/or ignored. It seems that some computers cannot supply enough power through their on-board USB ports to directly drive the transmitter. The use of a self-powered USB hub should be incorporated to ensure adequate power is available for the transmitter used to converse to the RoboScout. I've heard that other USB devices can also benefit from using a powered hub. In my case, I used a Kensington DomeHub model #K33118A seven-port powered USB 2.0 hub. You should be able to use almost any USB 1.1 or USB 2.0 hub as long as it is self-powered. Another benefit of using an external hub is that if anything went wrong with the control box that could damage the USB connection, it would most likely take out the USB hub instead of your computer. The hub is easily replaced where the built-in USB ports on your system are not.

Using the Virtual Remote and Writing Scripts

Now that the PC interface is connected to the host computer and the software is installed, we can launch the virtual remote application. It is just a program called **robot.exe** that the installation utility copied to your system. Once it starts, you'll



The PC remote control application.



see the virtual remote control on the screen. At this point, you can click on any of the buttons to move the robot around and control it. Most of the movement keys toggle the state. Click it once to start the motion and then again to stop it.

Opening the panel to run scripts isn't obvious. On the upper right side of the virtual remote (just under the three standard windows panel buttons), you will find a single button with two greater than signs (>>) which opens up the scripting window. This is where you can try individual commands or execute scripts. To try a single command, just enter it into the left of the "Exec Cmd" button, then press the button to execute the command. To run a script, you first have to click on the "Read File" button which opens up a file browsing window to select the script to load. Once

you select it and load it, you can then execute it at any time using the "Run Script" button. Pressing that button will execute (or re-run) the currently loaded script. The script will start out from whatever state the robot was in when you began the script. If you need the robot to start in a known state, then you should set that from within your script if you can (like speaker on/off), or manually for some commands that only toggle the state (like the light on the front of the robot).

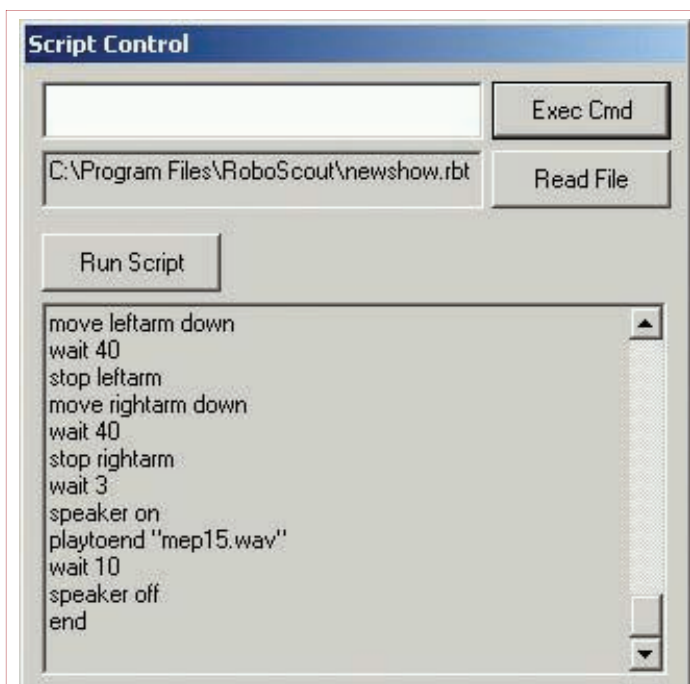
There is no built-in editor for creating the scripts for the robot. This isn't a problem since they are just straight text files which you can create and edit using Notepad. When working on scripts for the robot, I usually keep Notepad open to update the scripts, save the changes, and then reload and test the scripts with the RoboScout software.

So far, I haven't found a way through the scripting language to leverage any of the robot's built-in speech or to generate speech on-the-fly. All of the speech that you want to use in your script needs to be created by another method. For this, I have had excellent results using an application called TextAloud by NextUp. It lets you enter any text you want and the voice to use; with that, you can either perform the text to speech directly or write it to a .WAV file which works perfectly with the RoboScout scripting language. Each phrase that you want the robot to say should be created as separate .WAV files stored with your program script. With this, you can make RoboScout say anything you want it to. Using TextAloud, it doesn't matter if you can't use the built-in phrases.

Scripting Hints

Some commands like the **light change** should have a small delay using the **wait** command afterwards before sending the same command again. This will ensure they will be properly executed and none of them skipped. During testing, if I sent several **light change** commands in a row, some could be ignored and the light wasn't in the state I expected it to be in. Also, scripts work best when playing sounds independent of movement commands.

Scripting window for PC applications.





The naming of the movement commands is just odd. Unlike any normal convention I have seen, the script uses compass headings for specifying movement. The robot does not have a built-in compass so the actual compass heading is meaningless. It only makes sense if you are looking at a compass heading and reference north as forward.

Desired Robot Movement	Script Direction
Forward	North
Forward/Right	Northeast
Right	East
Backward/Right	Southeast
Backward	South
Backward/Left	Southwest
Left	West
Forward/Left	Northwest

Keep in mind that the RoboScout is an open loop system and movements are unfortunately not absolute or perfectly controlled. You can't directly specify a distance to move. Instead, you start the movement, wait, and then stop it. Most motion is started and dependent on delays written into each script. Delays may need to be adjusted for each particular robot and/or the speed of the host PC being used. Go through a few practice runs and eventually you can get it fine-tuned to work well enough.

Example RoboScout Script

Writing the actual script for the RoboScout is easy. Just use one command per line like the example program included here. It shows off some movement commands, changes the state of the light on the robot, and finally plays

List of Robot Commands for Scripting

Here is a list of all the available commands that can be used for writing scripts to control the RoboScout robot. Alternatives are shown in parentheses, separated by a vertical bar.

For example, `move head ( left | right )` means either move head left or move head right.

Optional elements are in square brackets. The default value is in *italics*.

move (lefthand | righthand)

Turns on the motor to open/close the specified hand. It cycles through opening and closing of the hand continuously until a **stop** command is issued.

move (leftarm | rightarm) (up | down)

Moves specified arm up or down until either a **stop** command is issued or the limit is reached.

move head (left | right)

Moves the head left or right until either a **stop** command is issued or the limit is reached.

move body (forward | backward)

Tilts the body forward or backward until either a **stop** command is issued or the limit is reached. When running scripts, the motion has "soft" limits which prevent the full range of movement that you'd expect.

move legs (north | northeast | east | southeast | south | southwest | west | northwest) [(fast | slow)]

Moves the robot in the specified direction and at the specified speed until a **stop** command is issued.

stop (lefthand | righthand)

Stops the movement of the specified hand.

stop (leftarm | rightarm)

Stops the movement of the specified arm.

stop body

Stops the tilting movement of the body.

stop legs

Stops moving the robot.

mode (remote | monitor | auto | sentry | greet)

Changes to the specified mode. Robot must be in remote mode to continue running scripts.

light change

Toggles the light intensity. Each toggle cycles between low, high, and off.

speaker (on | off)

Turns the speaker on or off. Speaker must be on in order to hear sounds from the **play** command.

video (on | off)

Turns the video on or off.

arms (lock | unlock)

Locks or unlocks the arms. When locked, the robot moves the arms to a special position to hold the serving trays. All following arm movement commands are ignored by the robot until the arms are unlocked again.

use "filename"

Loads the specified sound file as the default sound file. Useful for setting a global sound which is used repeatedly in a script.

play "filename"

Plays the specified sound file. Speaker must be on to hear the sound played through the robot.

play sound

Plays the default sound file. Speaker must be on to hear the sound played through the robot.

play stop

Stops the currently playing sound.

playtoend "filename"

Plays the specified file to the end. Any following script commands will not execute until playing sound file is complete. Speaker must be on to hear the sound played through the robot.

playtoend sound

Plays the default sound file to the end. Any following script commands will not execute until playing sound file is complete. Speaker must be on to hear the sound played through the robot.

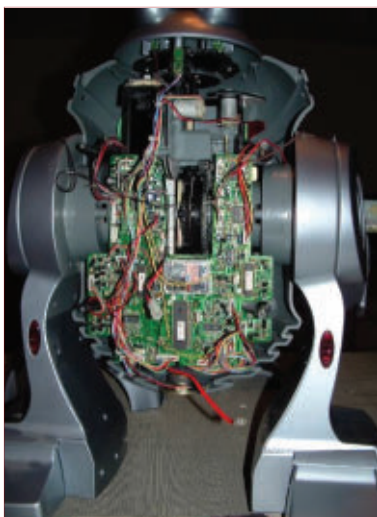
a .WAV file through the robot. Although there isn't a method for adding comments to the script, they are easy to follow.

speaker on	stop legs
move legs north	move body backward
light change	wait 7
wait 10	stop body
stop legs	light change
move legs east	wait 2
wait 8	light change
stop legs	play "functnl.wav"
move legs west	move body forward
wait 16	wait 7
stop legs	stop body
move legs east	light change
wait 8	end

If you are going to use the robot in a demonstration and plan on interacting (or perceived interaction) with it, there are a couple things you can do. Either use movements as visual cues, phrases, or sounds so the performer working with the robot can easily keep track of where things are in the script. I've used scripting to demo the robot for a school show and if the timing is right, it appears that you are having a conversation with the robot. If you decide to have the robot serve drinks, it is advisable to use closed cans since the robot can wobble a bit when moving. Otherwise, it may be a messy demo!

RoboScout Internals and Maintenance

So far, the RoboScout has been working flawlessly as long as I've had it. I've heard that some people have had to do some repairs to keep the robot running. Replacing the battery packs and the main drive belts seem to be the most common ailments. However, the RoboScout is not one of



the easiest robots to disassemble and work on. Luckily, there is some help online for those that need a hand keeping the robot in shape. There are some website links listed in the sidebar that show the internals of the robot and also how to tear it down to replace the drive belts. Besides that, there are a couple of simple things you can do if you run into trouble using the PC

interface. If the software complains that it cannot communicate with the control module and it was working before then, unplug the control module and wait a minute. Then, plug it back in so the OS can rediscover it. If the robot gets "stuck" in the mode where the arms are in the locked position (even after the power has been turned off and then back on again), you can usually get around it if you run the built-in DEMO mode which will clear it out. You can also try selecting a different video and control channel in case there is interference.

Conclusion

The RoboScout can provide hours of enjoyment as it is. Once you get bored with that or want it to do some new tricks, you can start writing your own scripts for the robot and use the PC to control all the actions. The built-in scripting has some limits, but carefully crafted scripts will allow you to do all sorts of new things with the robot. There are some other possible methods of programming and controlling the robot. One would be to write some programs to directly use some of the routines in the .dll files that the virtual remote program uses. The other is a bit more extreme where you would hack into the internals to replace the existing code or use a more powerful embedded processor as a new brain for the robot. That will be a topic for another day.

If you write any cool scripts for the robot or uncover any new tricks, I would love to hear from you. **SV**

Website References

Author's website
www.robotworkshop.com

Yahoo! Group for RoboScout
<http://groups.yahoo.com/group/RoboScout>

Sharper Image
www.sharperimage.com

RoboScout Internals
www.robin-hewitt.com/roboscout/index.html

RoboScout Drive Belt Replacement (see RoboScout section)
www.hero2000robots.com

RoboScout Videos
www.roboware.org/Gallery/v/robots/roboscout

RoboScout Demo
www.youtube.com/watch?v=6ghROMUNuTc&feature=related

TextAloud Text to Speech Utility
www.nextup.com



Then and NOW

A BRIEF LOOK AT SERVICE ROBOTICS

b y T o m C a r r o l l

Forty years ago, robotics consisted mainly of the industrial variety with a smattering of university, industrial specialty labs, and private experimenter's creations making the headlines. None of these robots were a finished manufacturer's product aimed at non-industrial applications but were more of a 'proof-of-concept' prototype. Certainly, many experimenters dreamed of an in-home robot to assist people in everyday chores, but most non-industrial machines of the '60s were aimed at surveillance uses or weapon systems. Isaac Asimov's many robot tales may have whet the appetites of those who wanted a robot to serve them, but few people had the faintest idea of how to design and develop a truly useful service robot.

Bipedal walkers were certainly not an option 40 years ago, so any experimentation centered on wheeled machines. However, today's market consists of literally hundreds of categories of robotics. Though still in the developmental stage and trailing an umbilical power and control cord, Twendy-one, a robot from Japan, (shown in **Figure 1**) can actually *physically* assist disabled people in their daily lives rather than following them around like an obedient dog as with most of the other 'home care bots' being developed. The 245 pound nurse robot may become available in a decade or so.

Let's look at the service category, which can include sub-categories of personal and consumer robots (toys, vacuum cleaner robots, etc.), military and defense robots, security robots, academic and research robots, healthcare and hospital robots, and a vast array of remotely-operated vehicles. Of these groups, there are more finite sections that may include mobile and non-mobile (fixed) versions. The mobile categories may include wheeled/tracked, airborne, sub sea, surface of the sea, and various other types. The non-mobile sub-category may include robotic subsystems built into

a larger system to add functionality.

New Battery Chemistries Allow Better Robot Designs

One area that I feel has allowed the tremendous growth in the service robots category is the vast improvement in battery technology. Almost all industrial robots tend to be mounted in a particular location on a production line or in a work cell, and derive their power (and some control) from hard-wired lines. Service robots are

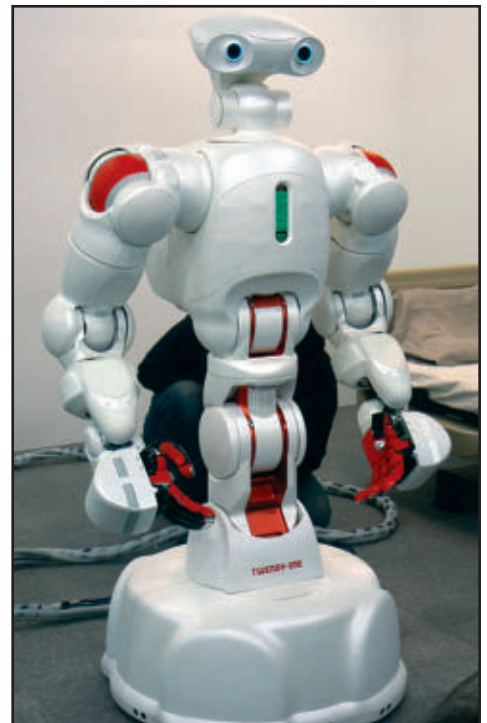


FIGURE 1. Twendy-one Assistant Robot.



FIGURE 2.
WowWee Rovio.

for the most part mobile devices, and must derive their electrical power from on-board batteries, a motor-generator, a fuel cell, or possibly solar panels, with batteries being the main source. Lead-acid, gelled electrolyte, and nickel-cadmium batteries were the only choices a decade or so ago. Nickel-metal hydride (NiMH) became popular

about 10 years ago as a higher power density and lighter source.

Today, many robot systems utilize lithium-ion (lithium cobalt oxide or Li-ion) batteries that offer two to four times the power density/mass capacity as NiMH batteries, two to three times faster charging, and a longer cycle life. Soon those swollen battery packs on the backs of larger bipedal robots and other service robots will shrink into smaller internal structures. The driving force behind the development of these batteries has been the 'green' trend.

Exactly What are Service Robots?

The International Federation of Robotics defines service robots in this way:

"Service robots have no strict internationally accepted definition, which, among other things, delimits them from other types of equipment, in particular, the manipulating industrial robot. IFR, however, has adopted a preliminary definition."

"A service robot is a robot which operates semi or fully autonomously to perform services useful to the well-being of humans and equipment, excluding manufacturing operations."

"With this definition, manipulating industrial robots could also be regarded as service robots, provided they are installed in non-manufacturing operations. Service robots may or may not be equipped with an arm structure as is the industrial robot. Often, but not always, the service robots are mobile. In some cases, service robots consist of a mobile platform on which one or several arms are attached and controlled in the same mode as the arms of the industrial robot."

"Because of their multitude of forms and structures, as well as application areas, service robots are not easy to define."

Well, that definition is about as clear as the answer to "what is a robot" with "I can't define what a robot is; I just know what a robot is when I see one." Besides, the third paragraph in the definition seems to center on any type of robot not used in manufacturing operations.

It's a Tough Sell for Robots Today

Despite the many new robot products making today's headlines — including the popular robot hamsters that hit last Christmas — the recession is hitting robot sales very hard in all industrialized countries. Recent articles about Japan's robotics

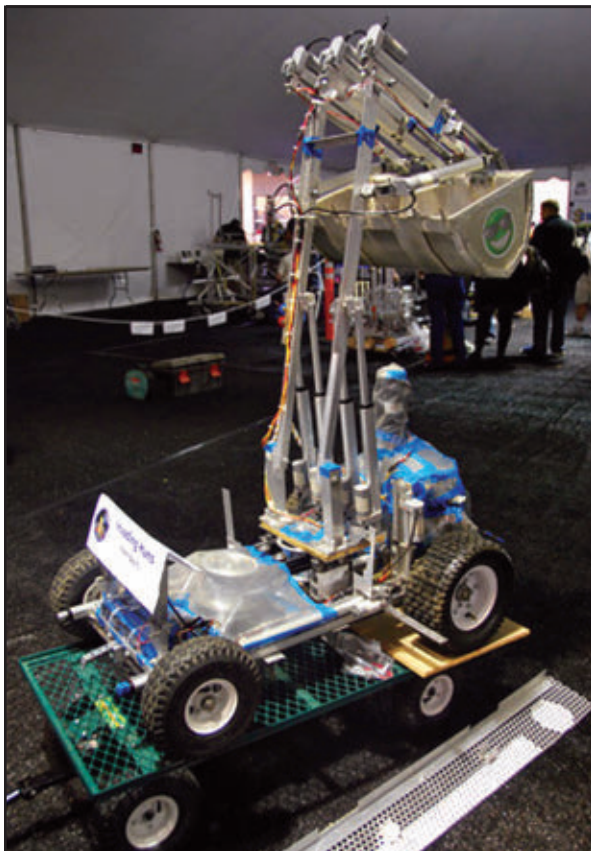


FIGURE 3. Lunar
Regolith Digging
Contestant.

industries — the hotbed of the world's market — have spoken about how hard this economic downturn has hit the robotics market.

The market for service robots is facing a gloomier outlook as personal spending for the latest gadgets has had to give way to things such as food, clothing, and shelter. The more sophisticated pet robots are cute and still adorable, but their lofty price tags make them a tough purchase to justify.

Robot Vacuum Cleaners Take Center Stage

Robot vacuum cleaners always seem to lead the assortment of service robots with iRobot's Roomba and their floor-washing Scooba at the head of the pack. (Don't forget their gutter cleaner series and robot pool sweepers.) And don't forget the robot lawn mowers.

The WowWee Rovio

The WowWee Rovio robot in **Figure 2** is an example of an affordable service robot that has capabilities of interest to most robot experimenters. It isn't a humanoid bipedal robot, a home appliance like a Roomba, or a kit or educational robot. It is a functional mini-security rover.

Initially, the Rovio had some problems with the battery dying before the robot had the chance to locate the charger base and recharge its batteries. Another problem mentioned by many experimenters was the camera's inability to see in a darkened room, and the tilt positions of the camera arm were not adequate. Early product reviews rated it anywhere from "I could never get it to work so I sent it back," to "I love it; it's a fantastic robot!" Recent reports indicate that the robot "has been greatly improved and most bugs have been eliminated" since its introduction in early 2008.

The 6-1/2 pound Rovio is priced at anywhere from \$180 to \$350, but most people have paid about \$210 for the robot. The Wi-Fi enabled webcam is touted as a "revolution in home exploration and telepresence" — the features that most robot experimenters and would-be hackers see as the most valuable in Rovio.

The NorthStar system enables the robot to know where it is, locate the position of other objects, and navigate accurately from place to

place under its own control. The web camera can be accessed from a computer, a cell phone, or even on the Internet from anywhere in the world. The Rovio's omni-directional wheels are another cool feature.

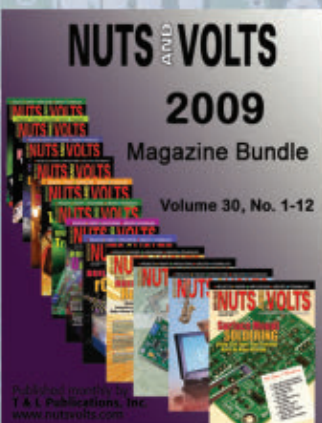
Robots to the Moon!

Competitions to develop robots to dig into the lunar soil have become popular around the world as this type of service robot keeps humans out of (potentially) harm's way during initial explorations. NASA's 2009 Regolith Excavation Challenge this past October was the third in a series and offered a total of \$750,000 in prizes. Held at NASA's Ames Research Center at Moffett Field, CA, 19 teams entered with the goal of loading at least 330 pounds of simulated moon soil called regolith into a container within 30 minutes. The remote-controlled robots could not be heavier than 176 pounds, including their on-board power source. Paul's Robotics of Worcester, MA took the \$500,000 first prize for digging and transferring 1,103 pounds. "Not bad for a half-hour's digging in the dirt." **Figure 3** is of one of the robot contestants.

Final Thoughts

I've touched on just a few examples of robot technology. Historically, service robotics separated from industrial robots on production lines and branched out as automated guided vehicles in factories. Today, that 'arena' has moved into homes, hospitals, care facilities, our oceans, the air, and even into space. It is service robots that all of us are striving to perfect — machines to serve Mankind. **SV**

*Tom Carroll can be reached at
TWCarroll@aol.com.*

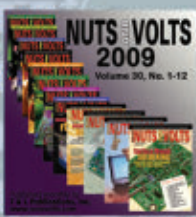


Choose your preferred format:
print or digital media.

We have just what you're looking for!

12 magazine bundle \$57.00
12 issues on CD-ROM \$24.95

Order today 800 783-4624
or online @ www.nutsvolts.com



ROBO-LINKS

BIPOM Electronics, Inc. Your One Source for μ Controller Systems
ARM, AVR, PIC, 8051, 680X, STAMP and others

μ Controller Boards
Peripherals
Development Tools
Emulators
Programmers
Robotics

Sensors
Dev. Training Kits
Instruments
Displays

GadgetPC ARM9 USB
Computer with Linux

www.bipom.com

WebCat+
Embedded Web Server

GPS MADE SIMPLE™

Linx Technologies

superbrightleds.com

Component LEDs - LED Bulbs - LED Products

Super Bright LEDs Inc. St. Louis, Missouri - USA superbrightleds.com

RobotShop.com

CAT CAN

www.catcan.com.tw

SMART SENSOR 2003 SMART SENSOR 4003 SMART SENSOR 2005 SMART SENSOR CONTROLLER

Pololu
Robotics & Electronics

3095 E. Patrick Ln. #12, Las Vegas, NV 89120
www.POLOLU.com 1-877-7-POLOLU

Motor, servo, and robot controllers
Sensors
Robots, motors, wheels, and more!

\$29.95 MaxSonar-EZ1
(MSRP)

High Performance
Ultrasonic Range Finder

- serial, analog voltage & pulse width outputs
- lowest power - 2mA
- narrow beam
- very easy to use!

www.maxbotix.com

The most comprehensive resource for news and links

THE ROBOT REPORT
TRACKING THE BUSINESS OF ROBOTICS

www.TheRobotReport.com

ALL ELECTRONICS CORPORATION
Electronic Parts & Supplies
Since 1967

**For the finest
in robots,
parts, and
services, go to
www.servo
magazine.com
and click on
Robo-Links.**

THE ORIGINAL SINCE 1994

PCB-POOL®
Beta LAYOUT

- Low Cost PCB prototypes
- Free laser SMT stencil with all Proto orders

WWW.PCB-POOL.COM

SERVO Magazine

2003/04 Volume 1, No. 1-12
2005 Volume 3, No. 1-12
2007 Volume 5, No. 1-12
2006 Volume 4, No. 1-12
2008 Volume 6, No. 1-12

5 CD-ROMs & Hat Special
Only \$ 109.00 (includes shipping)!

www.servomagazine.com

ADVERTISER INDEX

All Electronics Corp.	19, 82	Lynxmotion, Inc.	83	SuperBrightLEDs	82
AP Circuits	51	Maxbotix	82	Techno-Swap-Fest	31
BaneBots	29	Parallax, Inc.	7	The Robot Report	82
BiPOM Electronics	82	PCB Pool	35, 82	Vantec	6
CrustCrawler	15	Pololu Robotics & Electronics ..	63, 82	WeirdStuff Warehouse	19
Digilent	Back Cover	Robotis	2	Zagros Robotics, Inc.	57
Galaxy Far East Corp.	82	RobotShop, Inc	3, 82		
Linx Technologies	82	Solarbotics/HVW	7		

**To Advertise:
Call 951 371-8497**



The Lynxmotion Servo Erector Set Imagine it... Build it... Control it!

Featured Robot

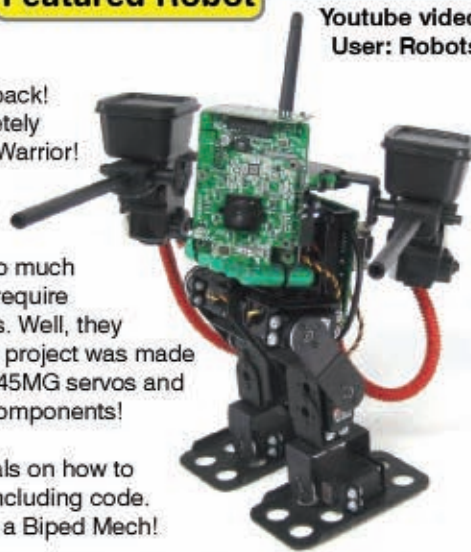
Biped BRAT does Mech Warfare!

Introducing the Hunchback!
The world's first completely functional biped Mech Warrior!

They said a BRAT based Mech couldn't be done. That it was too much payload. That it would require expensive digital servos. Well, they were wrong! This robot project was made from a BRAT with HS-645MG servos and standard off the shelf components!

We have created tutorials on how to make a Mech Warrior including code. Now anyone can make a Biped Mech!

Youtube videos
User: Robots7



Biped Nick



Biped Pete



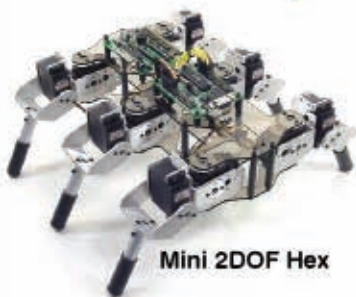
Biped Scout



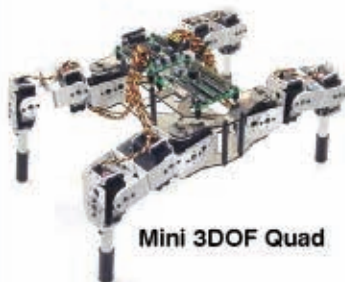
Biped 209



Walking Stick



Mini 2DOF Hex



Mini 3DOF Quad

With our popular Servo Erector Set you can easily build and control the robot of your dreams!

Our interchangeable aluminum brackets, hubs, and tubing make the ultimate in precision mechanical assemblies possible. The images here are just a sample of what can be built. The Bot Board II and SSC-32 provide powerful control options. Our Visual Sequencer program provides powerful PC, Basic Atom, or BS2 based control.



Bot Board II - \$24.95
Carrier for Atom / Pro, BS2, etc.
Servo and Logic power inputs.
5vdc 250mA LDO Regulator.
Buffered Speaker.
Sony PS2 game controller port.
3 Push button / LED interface.



SSC-32 - \$39.95
32 Channel Servo Controller.
Speed, Timed, or Group moves.
Servo and Logic power inputs.
5vdc 250mA LDO Regulator.
TTL or RS-232 Serial Comms.
No better SSC value anywhere!

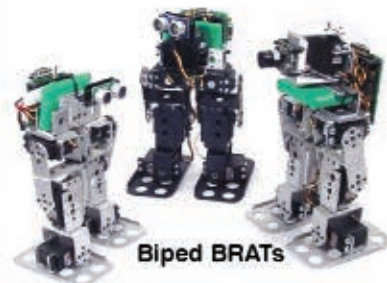
We also carry motors, wheels, hubs, batteries, chargers, servos, sensors, RC radios, pillow blocks, hardware, etc!



Visit our huge website to see our complete line of Aluminum and Lexan based robot kits, electronics, and mechanical components.



CH3-R Hexapod



Biped BRATs



Phoenix



Images represent a fraction of what can be made! www.lynxmotion.com The SES now has 157 unique components!

Face it. You've never been like the others.



We know how you feel.

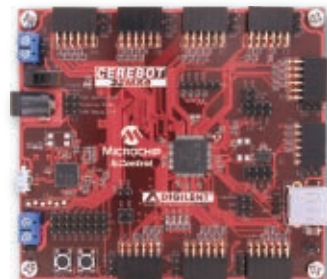
CEREBOTTM
32MX4

\$54

Enter the value code "servo2010a" on our website for this special introductory price

A powerful PIC32[®] based microcontroller board, ideal for robotic experimentation in C, C++, and Assembler.

- Microchip[®] PIC32MX460F512L
- 80 MHz 32-bit MIPS processor
- 512K Flash, 32K RAM
- Eight R/C servo connectors, two SPI ports, two I²C ports, two UARTs
- Nine 12-pin Pmod[™] Connectors
- Five 16-bit timers with 5 input capture and 5 PWM outputs
- 16 channel, 10-bit, 500kps A/D converter
- On-board USB Program / Debug Interface
- USB OTG Host / Device Capable
- USB Powered



Pmod[™]
Peripheral Modules

\$9.99 - \$29.99

Expand your designs

Pmods are a wide range of small, inexpensive & versatile I/O interface boards.



PmodJSTK
2-axis joystick
w/ 3 pushbuttons



PmodHBS
2A H-bridge
w/ feedback inputs



PmodENC
Rotary encoder module
w/ integral pushbutton



PmodMIC
Electret mic w/
compressor, & 12-bit A/D



PmodCLS
2-line character LCD
w/ serial interface



Digilent, Inc. is a leader in the design & manufacture of FPGA and microcontroller technologies. Our products can be found in over 100 countries and over 2000 universities worldwide. Visit our website for a wide assortment of these boards, as well as peripherals, reference designs, sample projects, tutorials, textbooks, and more.

www.digilentinc.com